An EA Process Template for PMs

Robert M. Flowe, Major, USAF
Defense Acquisition University

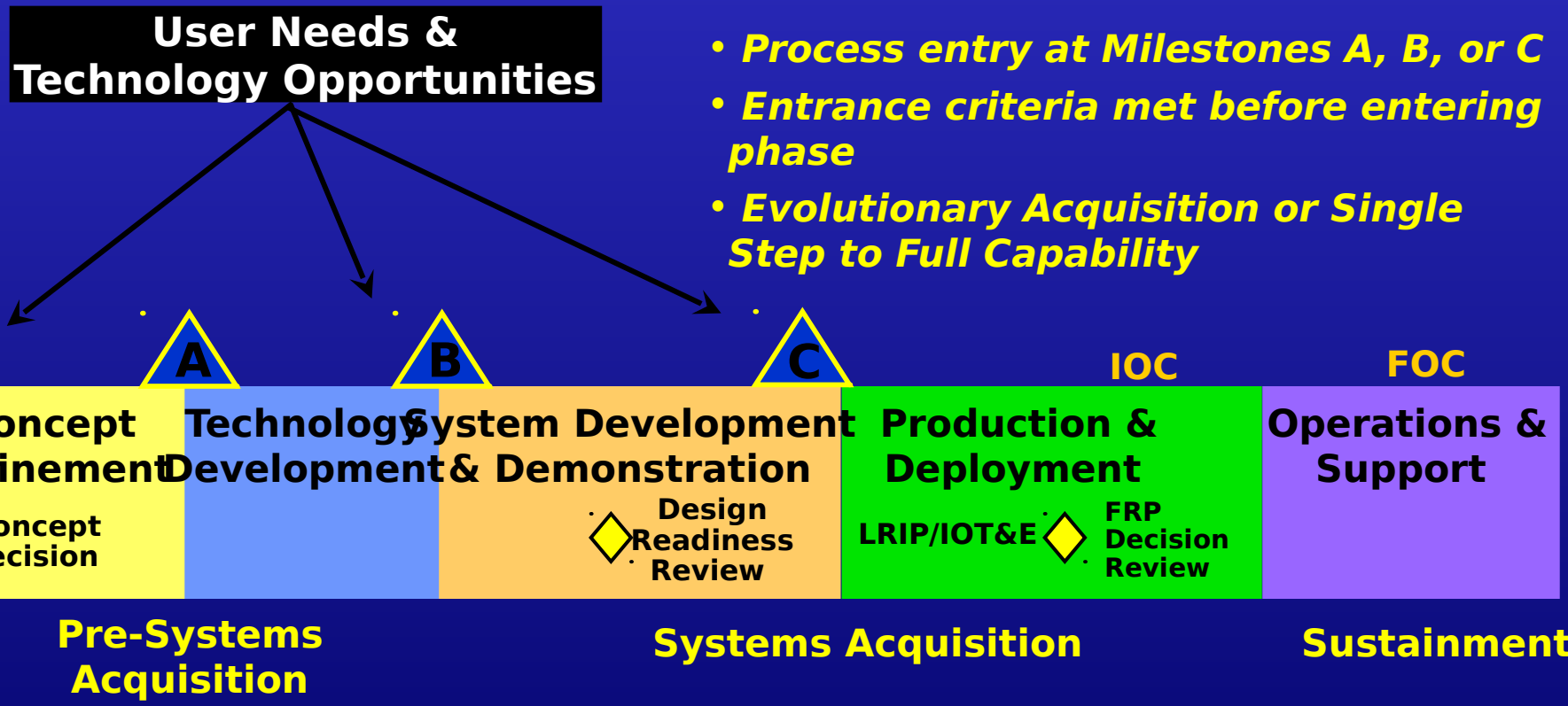
This presentation represents the views of the author, and does not represent the views of the Department of Defense, the Air Force, or Defense Acquisition University

Agenda

- The “New” Acquisition Environment
- Definitions: Evolutionary Acquisition & Spiral Development
- Why an EA Process Template?
- Key Features of this Template
- Process Description
- Special Interest Items:
  - Contracting Implications
  - Logistics Implications
  - Test and Evaluation Implications

The Defense Acquisition Management Framework

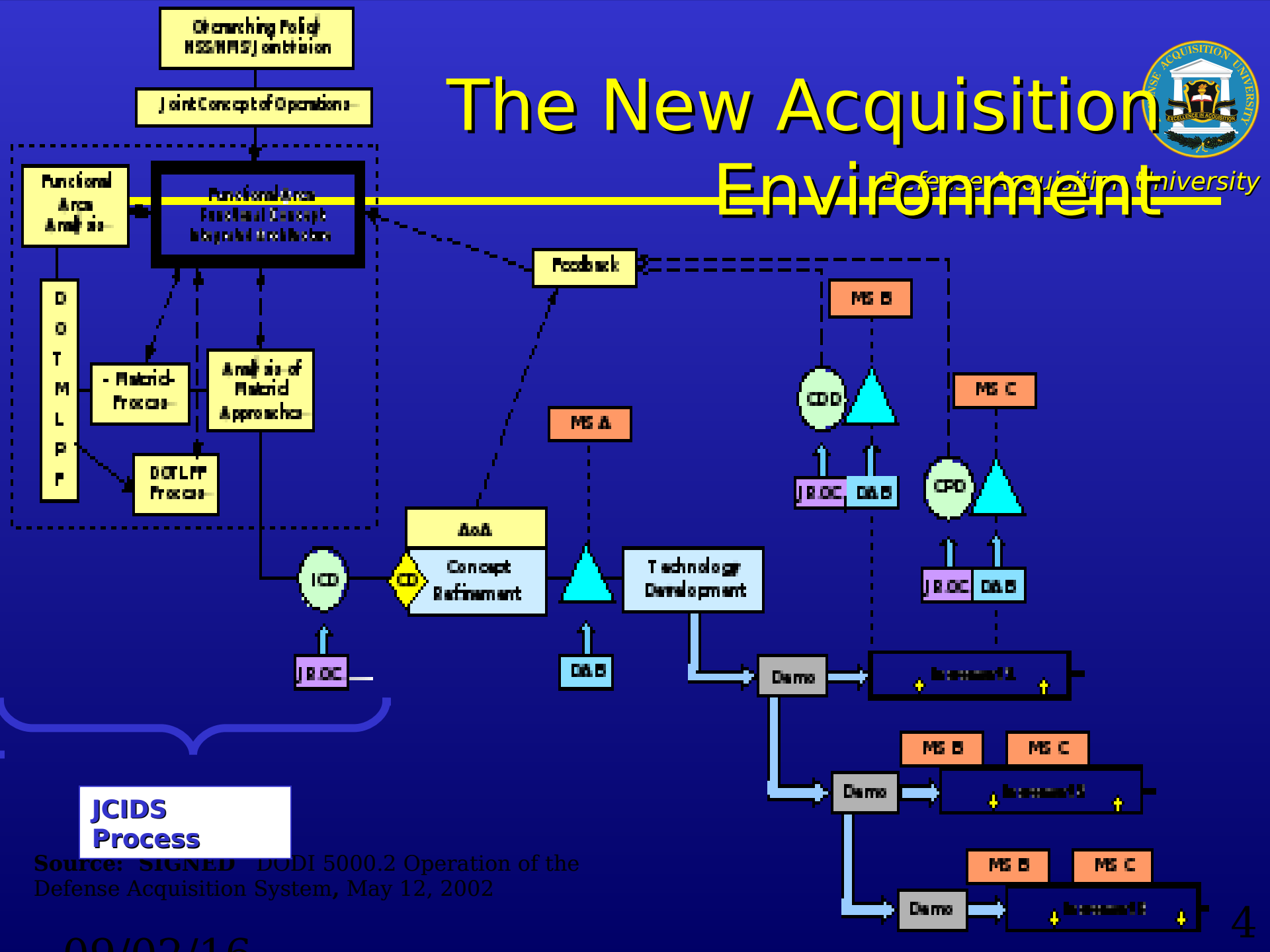
Defense Acquisition University



- ***Process entry at Milestones A, B, or C***
- ***Entrance criteria met before entering phase***
- ***Evolutionary Acquisition or Single Step to Full Capability***

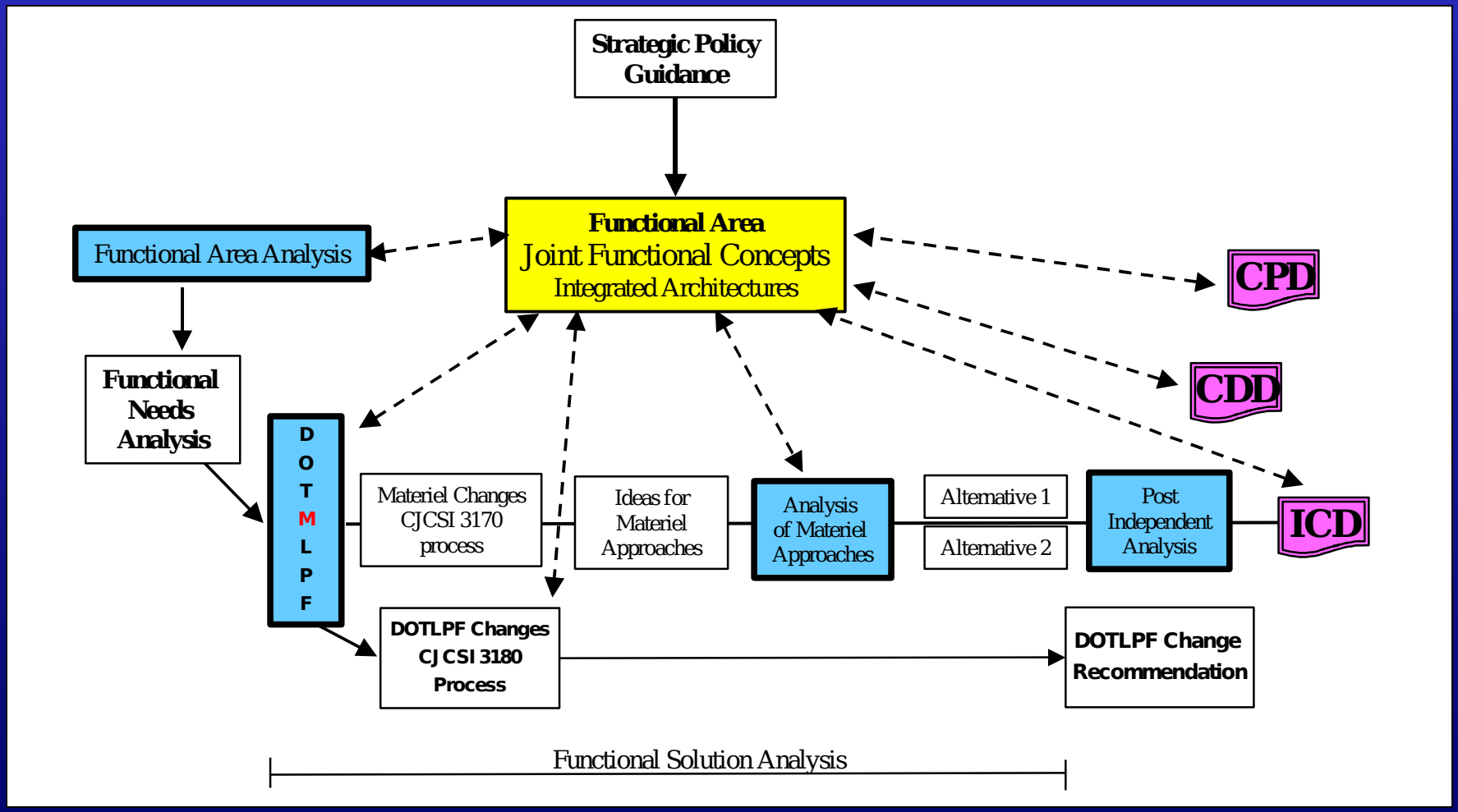
Fig. 1, DoDI 5000.2

The New Acquisition Environment



Source: "SIGNED" DODI 5000.2 Operation of the Defense Acquisition System, May 12, 2002

JCIDS Process



Source: CJCSM 3170.01M FLAG STAFFING DRAFT April 2003

Important DoD 5000 References

Defense Acquisition University

- **DoDD 5000.1, Paragraph 4.3.1:** There is no *one* best way to structure an acquisition program to accomplish the objective of the Defense Acquisition System. MDAs and PMs shall tailor program strategies and oversight … to fit the particular conditions of that program, consistent with applicable laws and regulations and the time-sensitivity of the capability need.
- **DoDI 5000.2, Paragraph 3.1.1:** Consistent with DoDD 5000.1 the program manager (PM) and the MDA shall exercise discretion and prudent business judgment to structure a tailored, responsive, and innovative program.

You Are Empowered To Do What Makes Sense!

Evolutionary Acquisition

- Evolutionary acquisition is the preferred DoD strategy for rapid acquisition of mature technology for the user.
  - delivers capability in increments,
  - recognizes the need for future capability improvements
  - objective is to balance needs and available capability with resources, and to put capability into the hands of the user quickly
  - success depends on consistent and continuous
    - definition of requirements,
    - maturation of technologies
    - disciplined development and production of systems
  - provide increasing capability towards a materiel concept.

Source: DoDI 5000.2, *Operation of the Acquisition System*, signed 12 May 2003

Why EA?

- Two Schools of Thought:
  - Better-Faster-Cheaper
  - Acquisition “Agility”
- Not totally complementary concepts
  - Establish different goals and expectations
  - These differences will be reflected in the acquisition strategy
- Herein lies a fundamental risk!
  - Make sure you know which view your stakeholders advocate!

Better-Faster-Cheaper

Defense Acquisition University

- Relies on mature technologies to reduce risk
  - Transfers development risk to others
  - Focus on integration of demonstrated capabilities versus development of new capabilities
  - Dependent upon technologies developed and matured by others
- Requires managing user expectations to accept increments of mature versus leading-edge capabilities
- Most successful where robust commercial market mirrors military requirements

Agile Acquisition

- The "Achilles Heel" of acquisition has been requirements change
- Lengthy, Linear-Sequential acquisition approaches very vulnerable to these changes
  - Creates unplanned rework, redesign effort
  - Manifests as cost/schedule growth
  - Perpetuates inefficient "crisis management" mode
- Evolutionary acquisition is more "agile"
  - Accepts requirements change as natural
  - Establishes controlled processes to deal with change
  - Minimizes "crisis management" diseconomies
  - Allows prompt convergence to the "right solution"

When to Apply "Agile" EA

Defense Acquisition University

- Requirements volatility or uncertainty
  - Rapidly-evolving threat
  - Rapidly-evolving doctrine, procedures, and/or requirements,
  - Functional or performance-based requirements
  - Rapidly-evolving technology
- Resource volatility or uncertainty

Given the prevalence of these conditions, it is clear why EA is designated the "preferred approach"

Why an EA Process Template?

Defense Acquisition University

- Most EA literature and policy discusses EA in general terms
- PMs are left to figure out how to implement EA for their programs
- Not much “how to” information available
- General anxiety about how to proceed

This EA process template describes key features and functions that must exist, in some form, for EA to work. Implementation details will vary by program

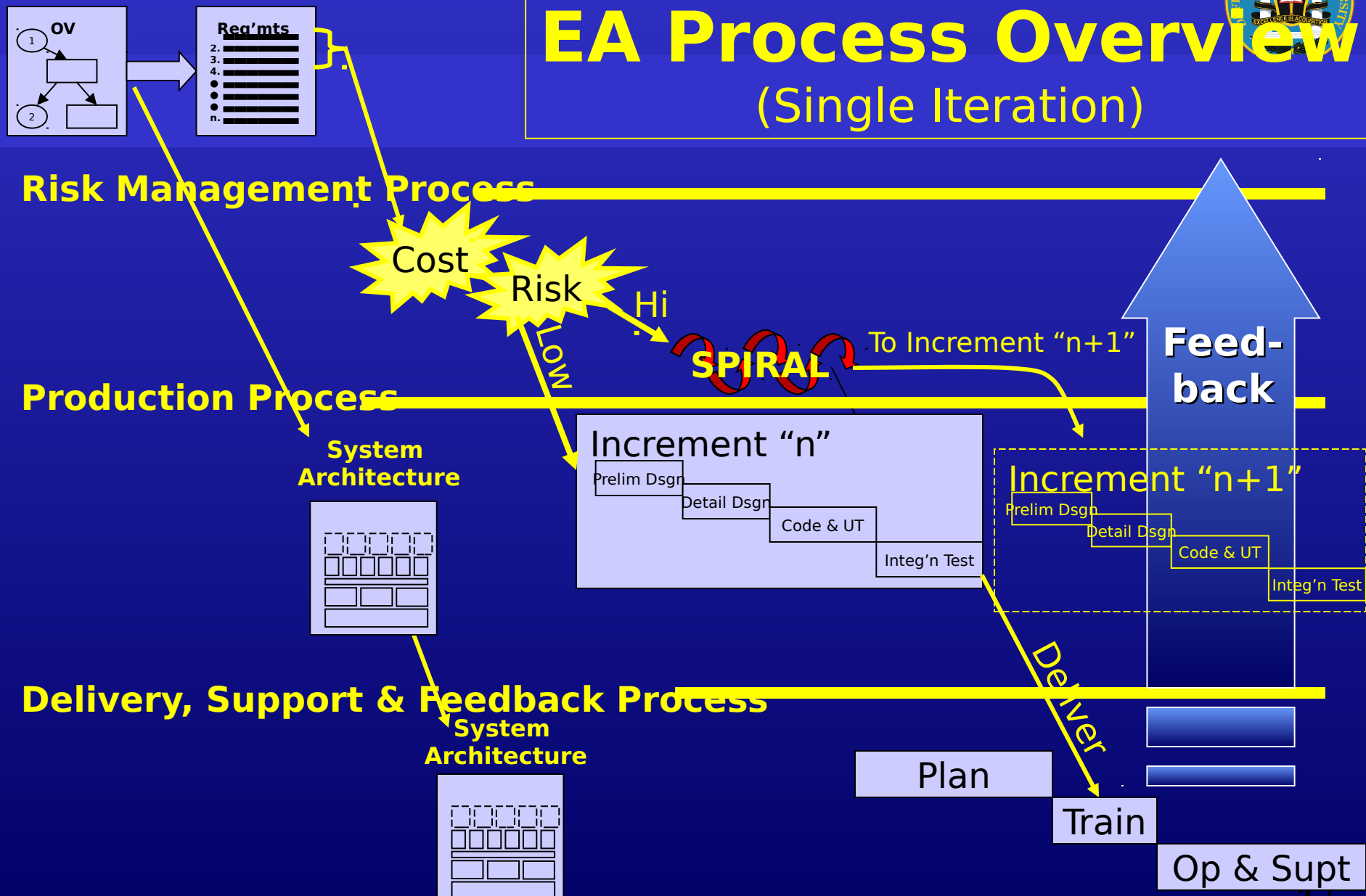
Key Features of This EA Approach

Defense Acquisition University

- Acquirer, User, and other stakeholders collaborate in requirements definition & domain analysis to develop an integrated program-level architecture
- Operational, System, and Technical Views used to identify, validate, prioritize, and order requirements
- Stakeholders participate in the continuous management of cost and development risk
- Frequent feedback between acquirer and all stakeholders throughout development & support
- Four major concurrent processes:
  - Requirements Management
  - Risk Management
  - Production (including test & evaluation)
  - Delivery, Support & Feedback

EA Process Overview

(Single Iteration)



Investment in Architecture

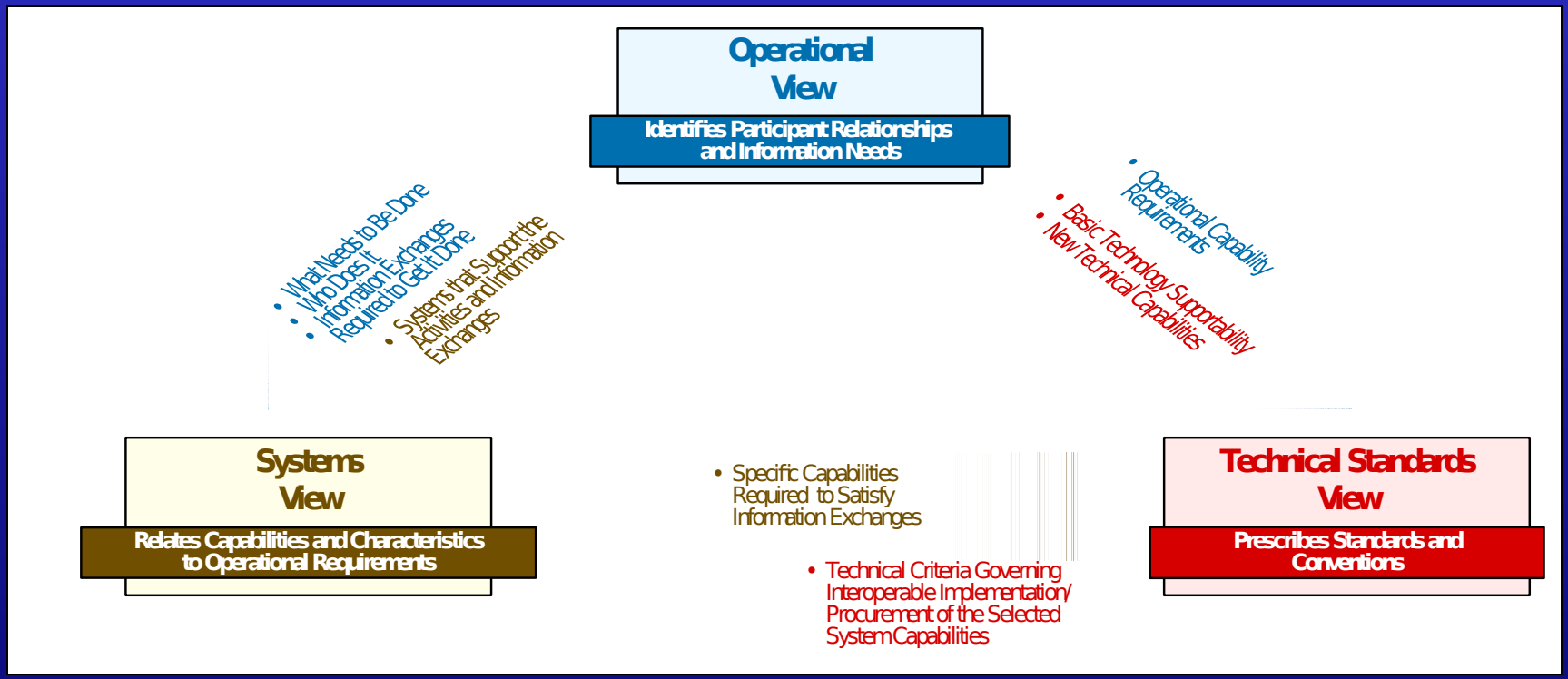
Requirements and Architecture

Defense Acquisition University

- Requirements don't "just happen". They reflect functions to be performed to provide capabilities that fulfill operational needs
- Requirements must therefore be mapped to operational needs (this is not new)
- The acquirer and user must share a common understanding of the operational domain as it relates to the program
- The mechanism for establishing and maintaining this understanding is the integrated program-level architecture and its related views

Integrated Architecture

Defense Acquisition University



Architecture: the structure of components, their relationships, and the principles and guidelines governing their design and evolution over time.

DoD Integrated Architecture Panel, 1995, based on IEEE STD 610.12

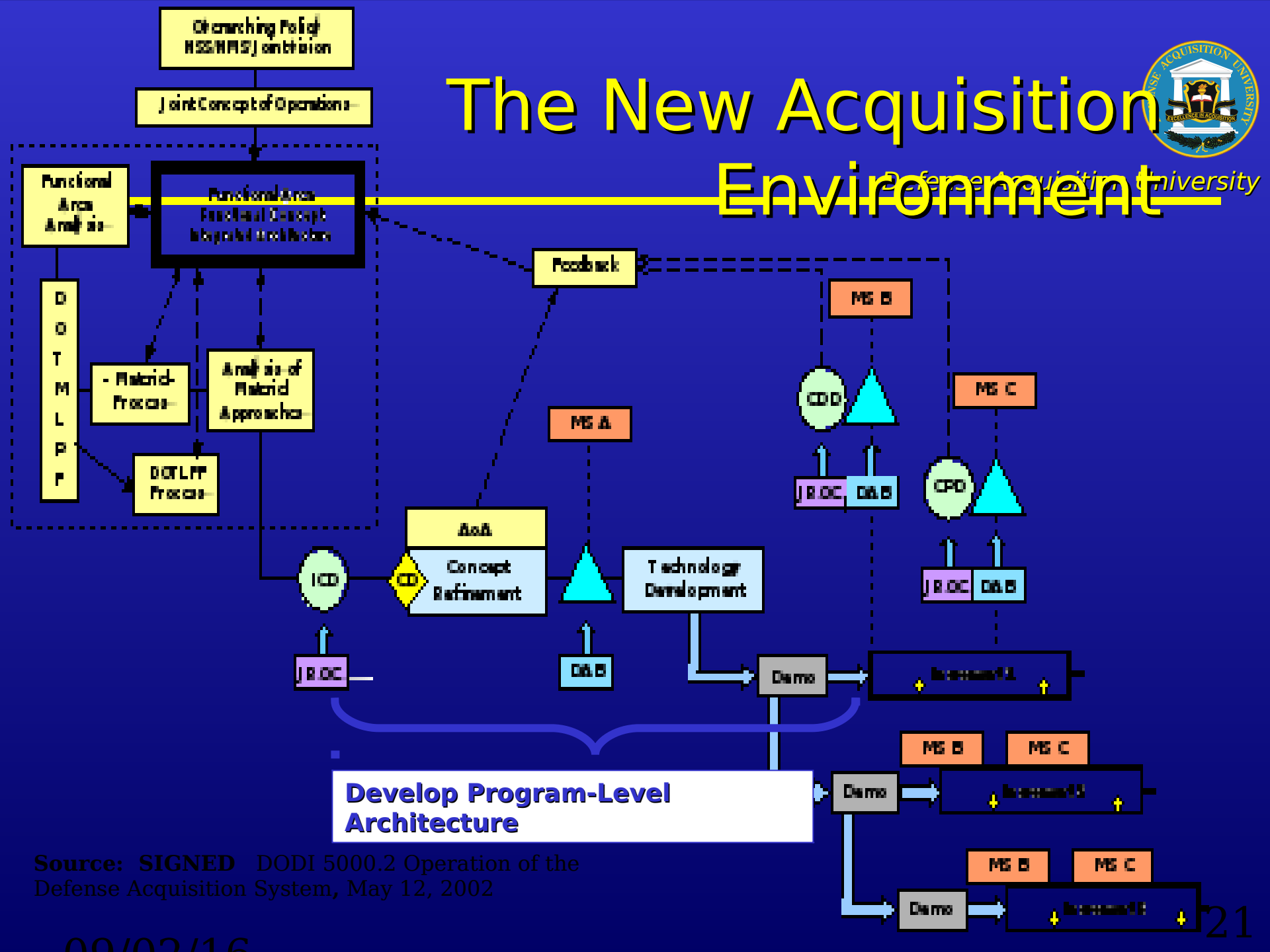
Source: DoD Architecture Framework, Version 1.0, Volume 1 [Final Draft], January 2003

Program Level Architecture

Defense Acquisition University

- Architectures exist at many levels
- Enterprise-wide architectures address the “Big Picture”
  - Joint Operational, Technical Architectures
  - Global Information Grid
- These constrain design options to enhance commonality, reuse, and interoperability
- These are insufficiently detailed to inform the development of program-level requirements
- A program-level integrated architecture must be developed for program-specific requirements
- The program-level architecture must be consistent with higher-level architectures
- This investment in program-level architecture should occur before Milestone B

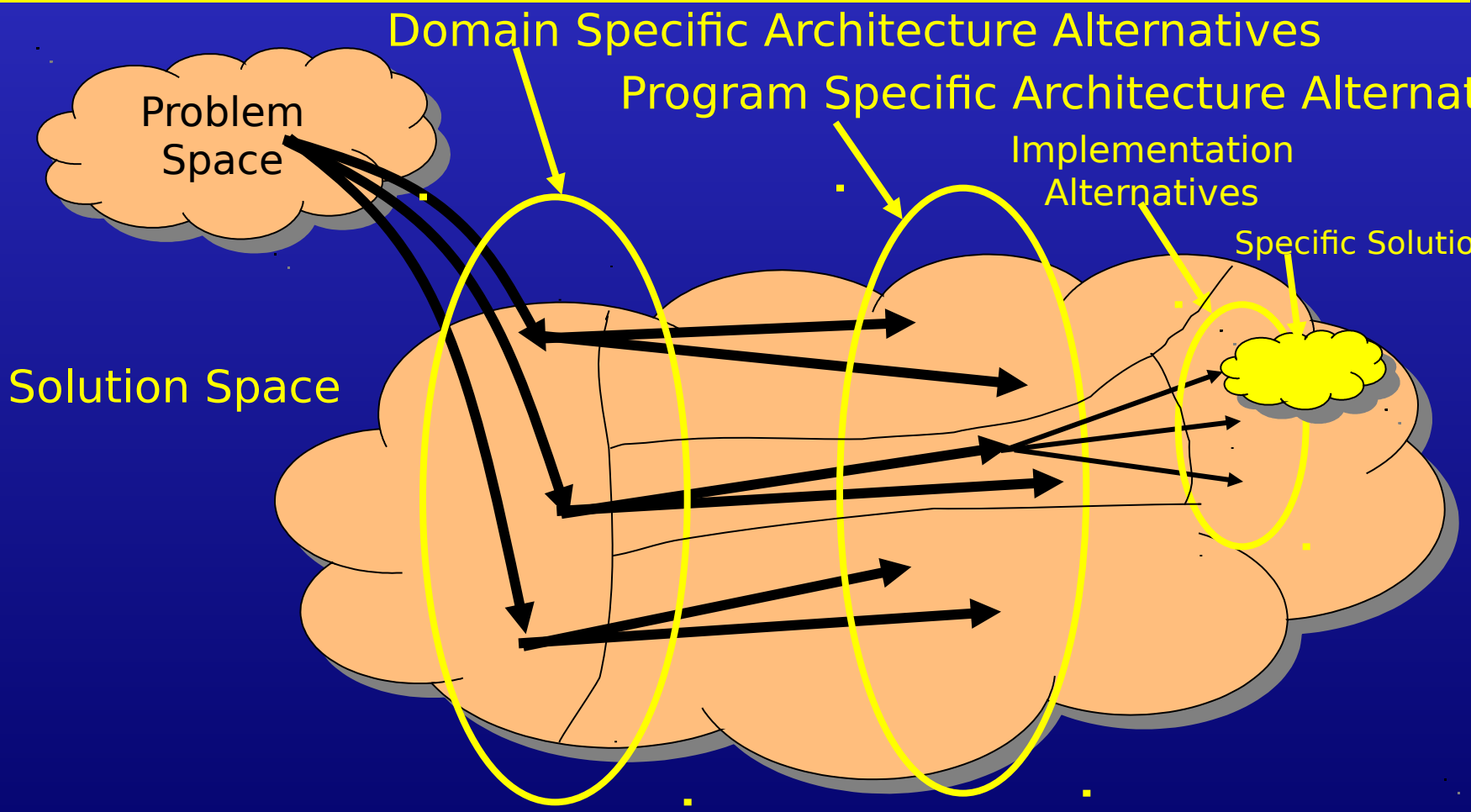
The New Acquisition Environment



Source: SIGNED DODI 5000.2 Operation of the Defense Acquisition System, May 12, 2002

Domain and Program-Specific Architectures

Defense Acquisition University

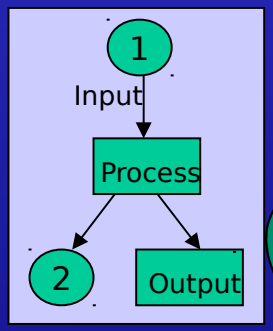


Based upon USC-CSE DSSA course, January 2001

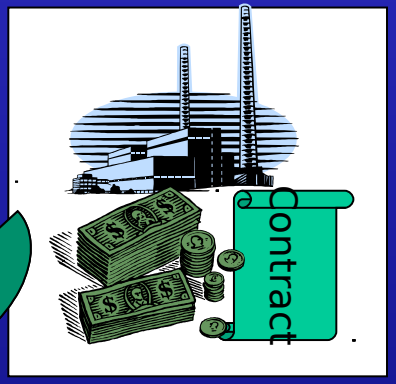
Views of the Solution Space

Defense Acquisition University

Operational View

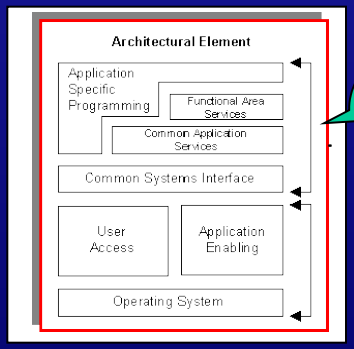


Business View



Solution

Systems View



Technical View

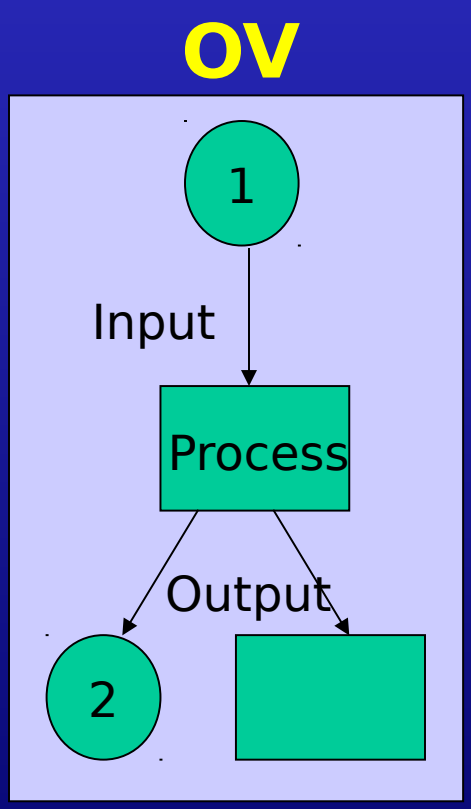
Specs/Stds

Also consider:

- Logistics view
- Security view
- Other views

Operational View

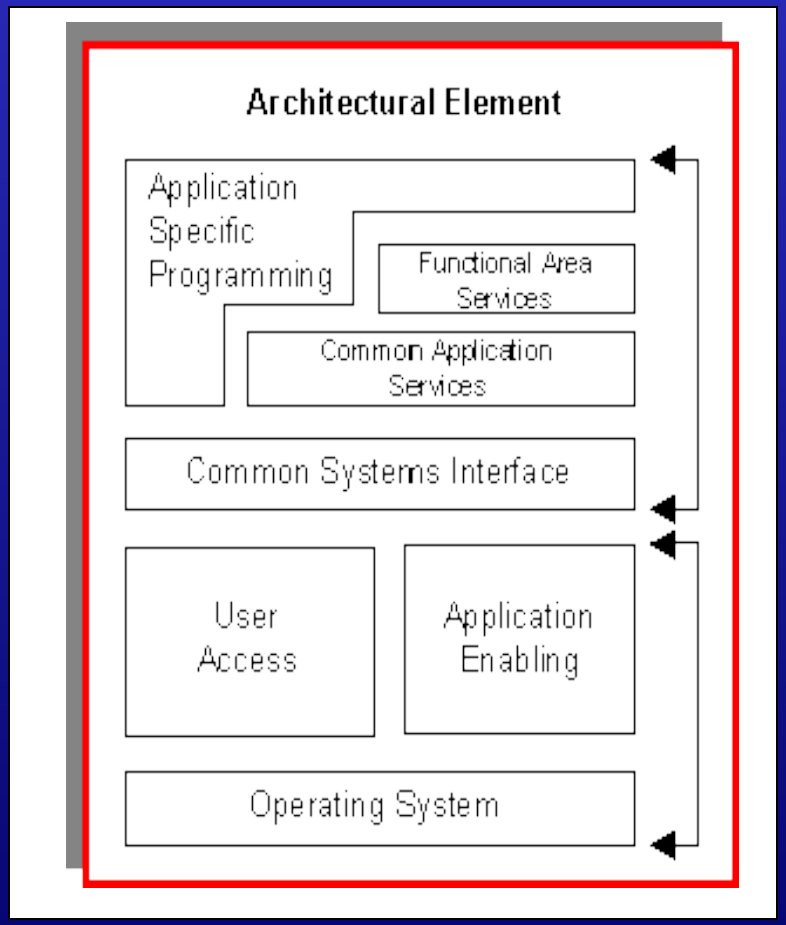
Defense Acquisition University



- Represents how the User operates
  - Current and future doctrine, tactics, techniques, & procedures.
  - Time-phased: describes required capability growth over time
- Focused on User, facilitated by Acquisition, S&T Communities
- Coupled with domain modeling to allow first-order tradeoffs on capability vs. cost vs. time
  - Balances "What is Needed" with "What is Possible"
- OV must be continuously updated and validated by User, endorsed by acquirer
  - Both must "own" the OV and related Current View (CV) as they evolve over time

The OV is a key instrument for achieving and maintaining acquirer/user/stakeholder understanding

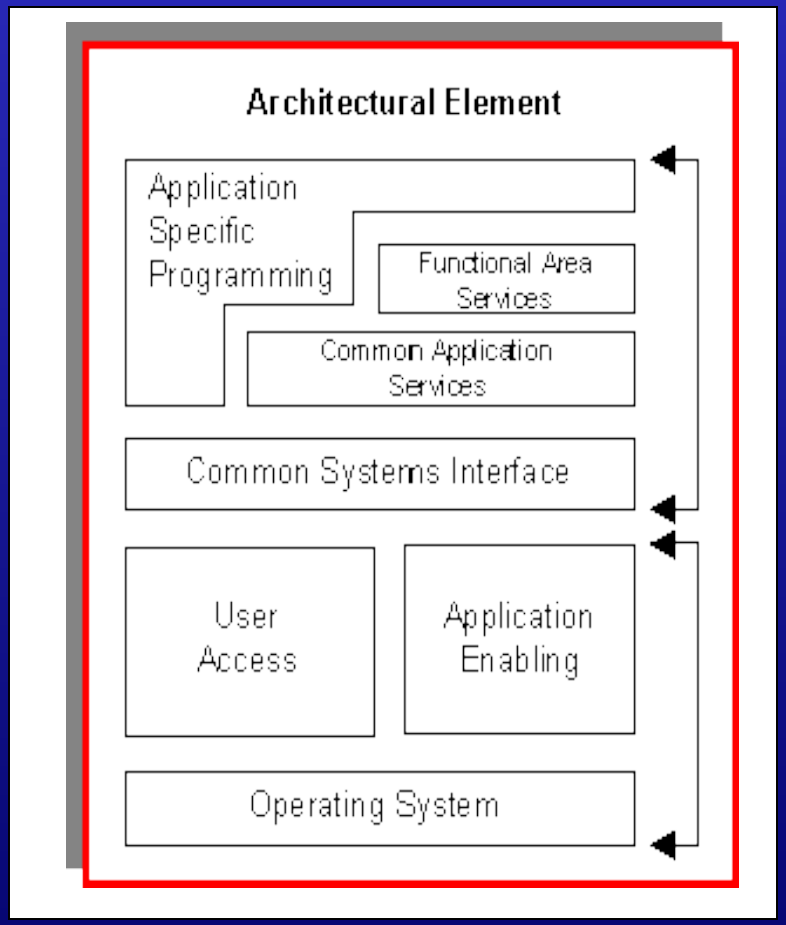
Systems View



- SV strongly influenced by the OV
  - Output of domain analysis & modeling
- SV describes functional "boundaries" of the program, interaction rules, and key interfaces
- Provides framework within which functional capability will operate and evolve
- SV reflects goals of flexibility, scalability, interoperability, robustness, security, etc.
- Establish or adopt an enterprise SV
  - Standard architectures exist for some domains
  - These enhance interoperability, facilitate reuse, and reduce risk

Graphic: Barry M. Horowitz, Ph.D.ESC-TR-94-208, September 1994

Systems View



- SV also influenced by requirements allocation
  - Grouping of dependent functions define functional modules
  - Required interaction between functional modules defines interface requirements
  - Aggregated functional interactions drives system-level design
    - Capacity, throughput, bandwidth, scale, processor speed, network topology, etc.

Graphic: Barry M. Horowitz, Ph.D.ESC-TR-94-208, September 1994

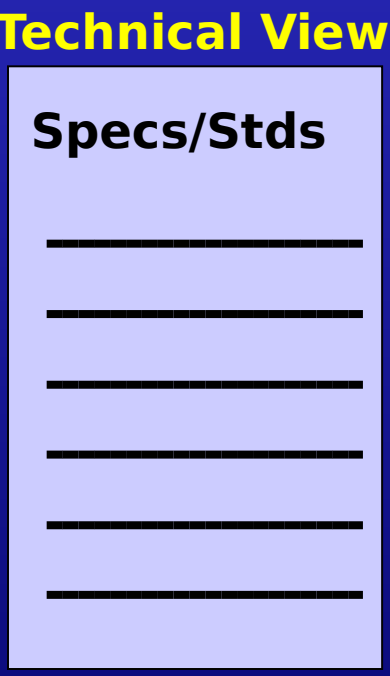
Executable Systems Architecture as a Deliverable

- The systems view of the integrated program-level architecture can be implemented as an executable product
- The process of transforming an abstract "view" into an executable product introduces rigor into the design process
  - Identify flaws in the concept and design
  - Allows demonstration of system-level functionality
  - Enables modeling of system capability over the lifecycle
  - Creates a reference to which others can design

Executable Systems Architecture as a Deliverable

- Modular, scalable, open systems architecture minimize integration and support cost over life cycle
  - Core functionality & services allocated to the architecture
  - Mission-specific functionality allocated to modular elements
  - Functional modules can be added and changed without extensive impact to architecture or other elements.
  - Support for core will be similar across configurations
  - Unique support requirements limited to functional modules
  - Standard interfaces minimize retrofit/upgrade cost
- Keep architecture distinct from functionality
  - System architect maintains architecture and imposes enterprise rules on functional elements
  - Maintaining “separation of powers” reduces likelihood of integrator monopoly

Technical View

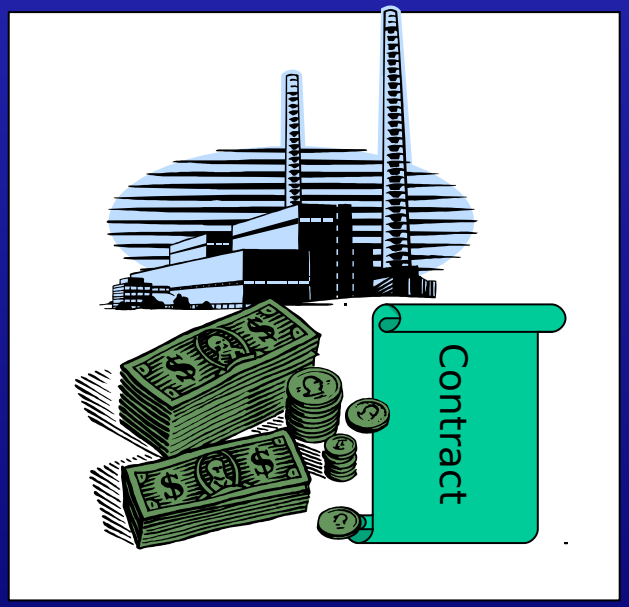


- Technical view of the integrated program-level architecture
- Describes the specifications and standards that apply to the domain for which the solution is an instance
  - Tailored as appropriate, with an eye toward maintaining interoperability
- Adherence to technical architectural standards reduces the design/engineering tradespace
  - Speeds design decisions
  - Promotes reuse
  - Enhances (but does not guarantee) interoperability
- Tradeoff: flexibility and performance

Business View

Defense Acquisition University

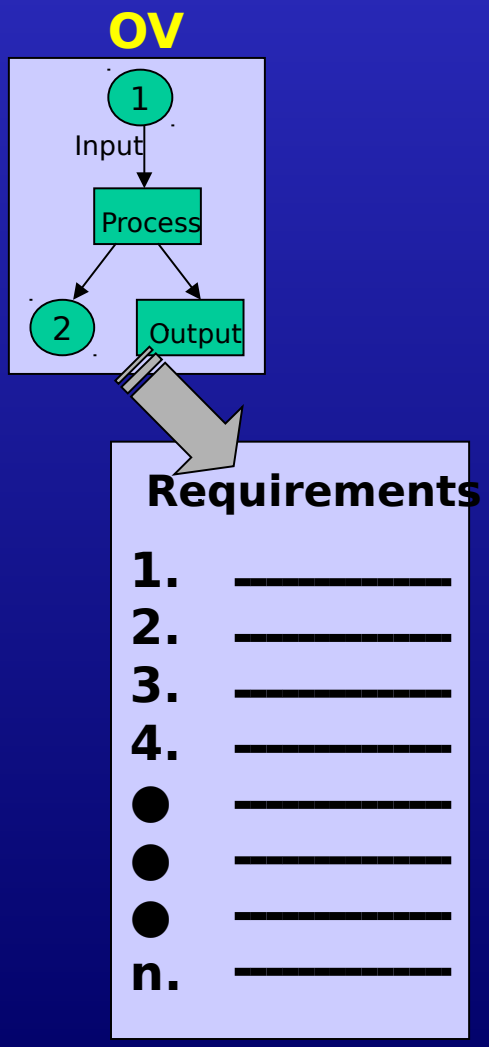
Business View



- Business View (for lack of a better term) provides insight into the organizational requirements and constraints of the enterprise
- Illuminates issues relating to
  - Acquisition strategy
  - Contract structures
  - Resource streams
    - Money
    - Personnel
  - Political issues
- Highlights the integration and interoperability requirements of these processes

Deriving Requirements from Architectures

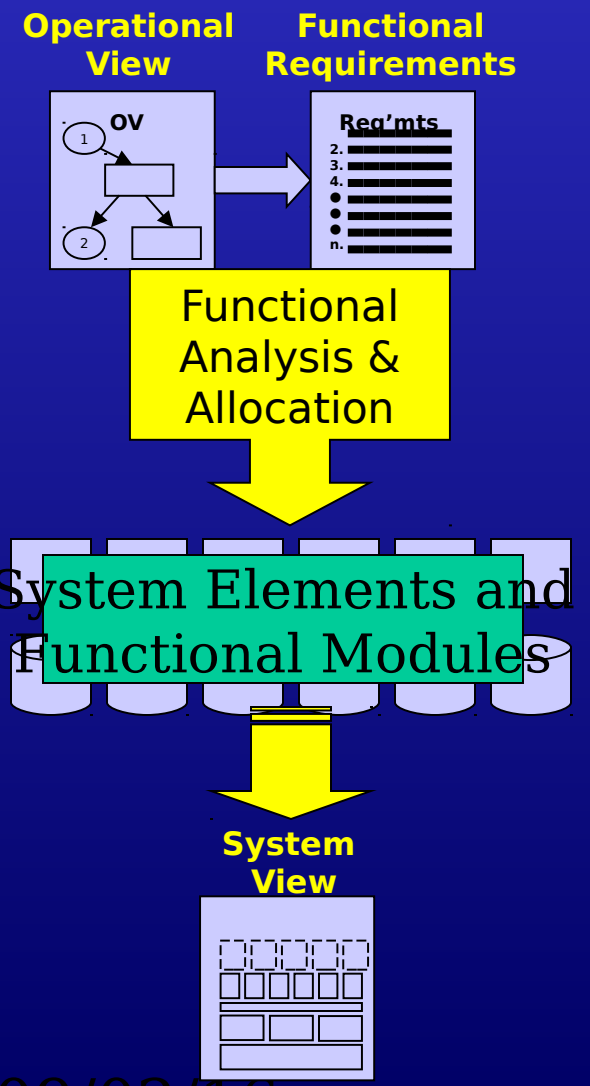
Defense Acquisition University



- OV describes organizational processes, inputs and outputs
  - This suggests functional capabilities required
    - What the system must do
    - Prioritized in order of operational relevance & timing
- Operational scenario analysis is useful
  - Users “walk through” their missions
  - Helps identify priorities, context, conditions, requirements, and functional dependencies

Functional Allocation

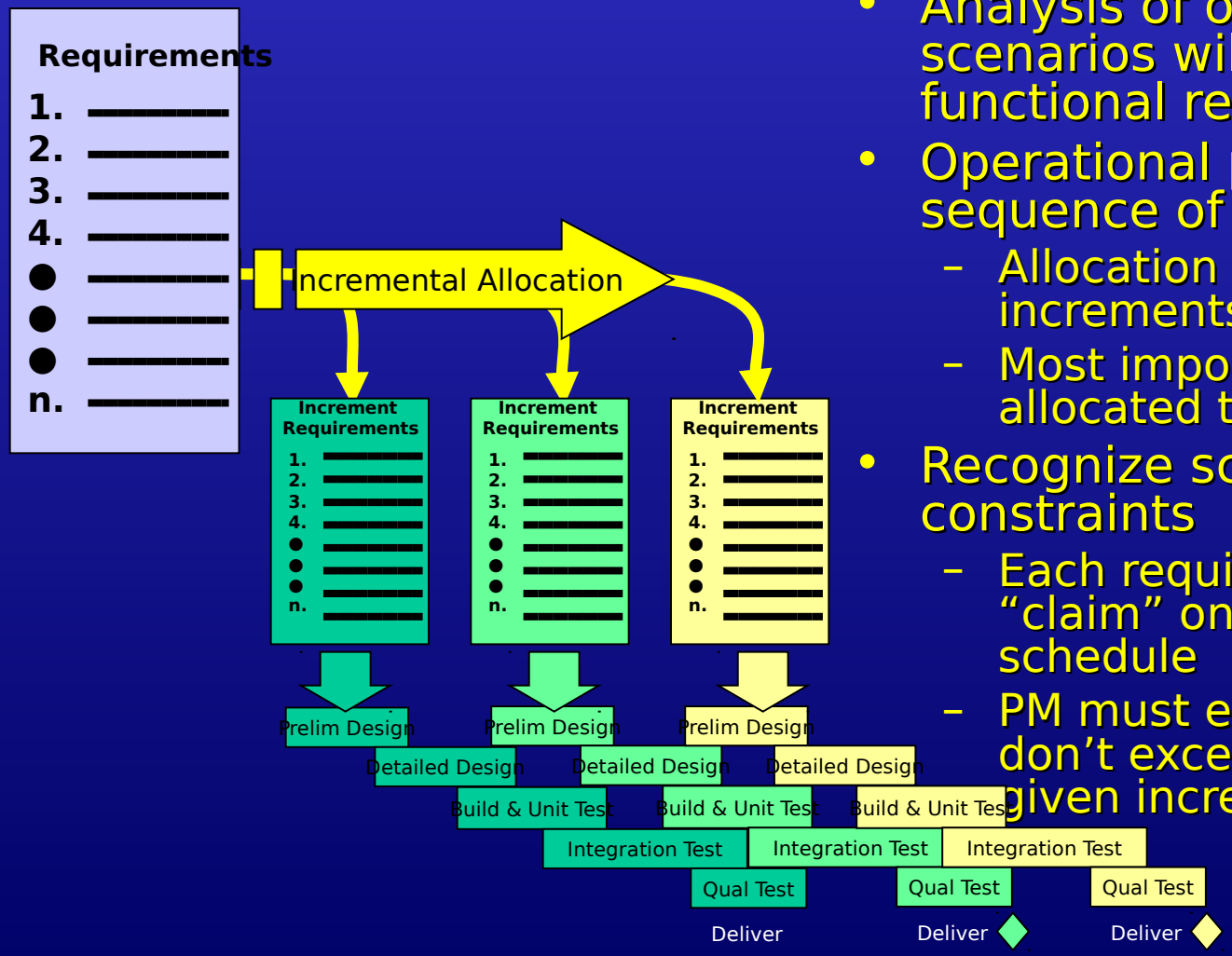
Defense Acquisition University



- Dependencies exist between requirements
  - Group requirements based upon functional coherence and dependence
  - Allocate similar/related requirements to functional modules—"functional allocation"
  - Requirements common to multiple modules should be allocated to the system as common services
- Functional allocation informs the development of the systems view of the integrated architecture
  - Distinguish functional elements from system services

Incremental Allocation

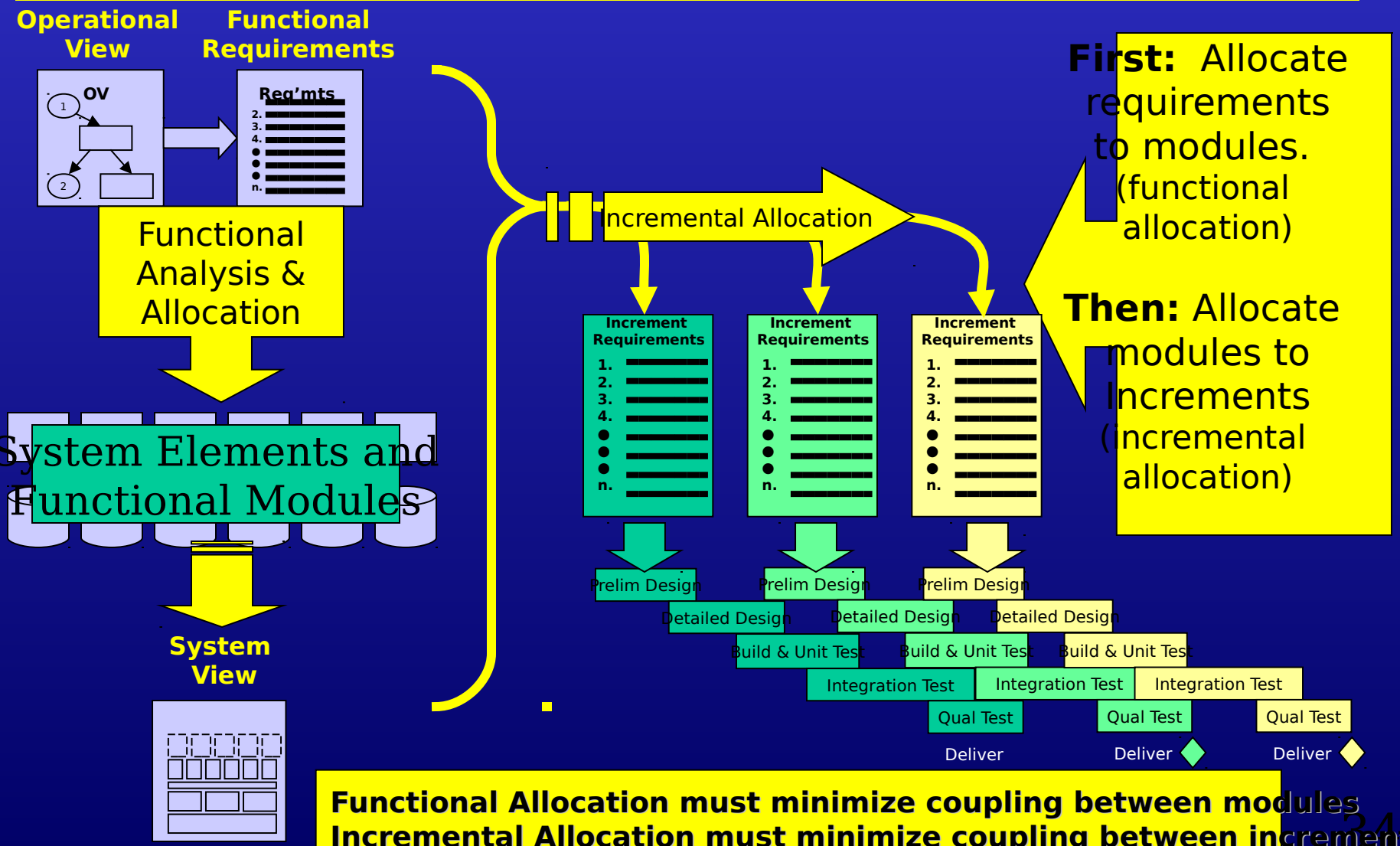
Defense Acquisition University



- Analysis of operational scenarios will help prioritize functional requirements
- Operational priority informs the sequence of implementation
  - Allocation of functions to increments
  - Most important functions allocated to early increments
- Recognize schedule and budget constraints
  - Each requirement represents a "claim" on resources: budget & schedule
  - PM must ensure commitments don't exceed resources for any given increment

Functional and Incremental Allocation

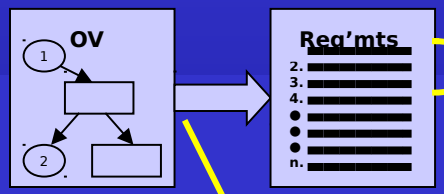
Defense Acquisition University



Process: Requirements Management

EA Process Overview

(Single Iteration)



Requirements Management Process

Risk Management Process

Cost

Risk

Hi

Low

SPIRAL

To Increment "n+1"

Feed-back

Production Process

System Architecture

Increment "n"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Increment "n+1"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Deliver

Delivery, Support & Feedback Process

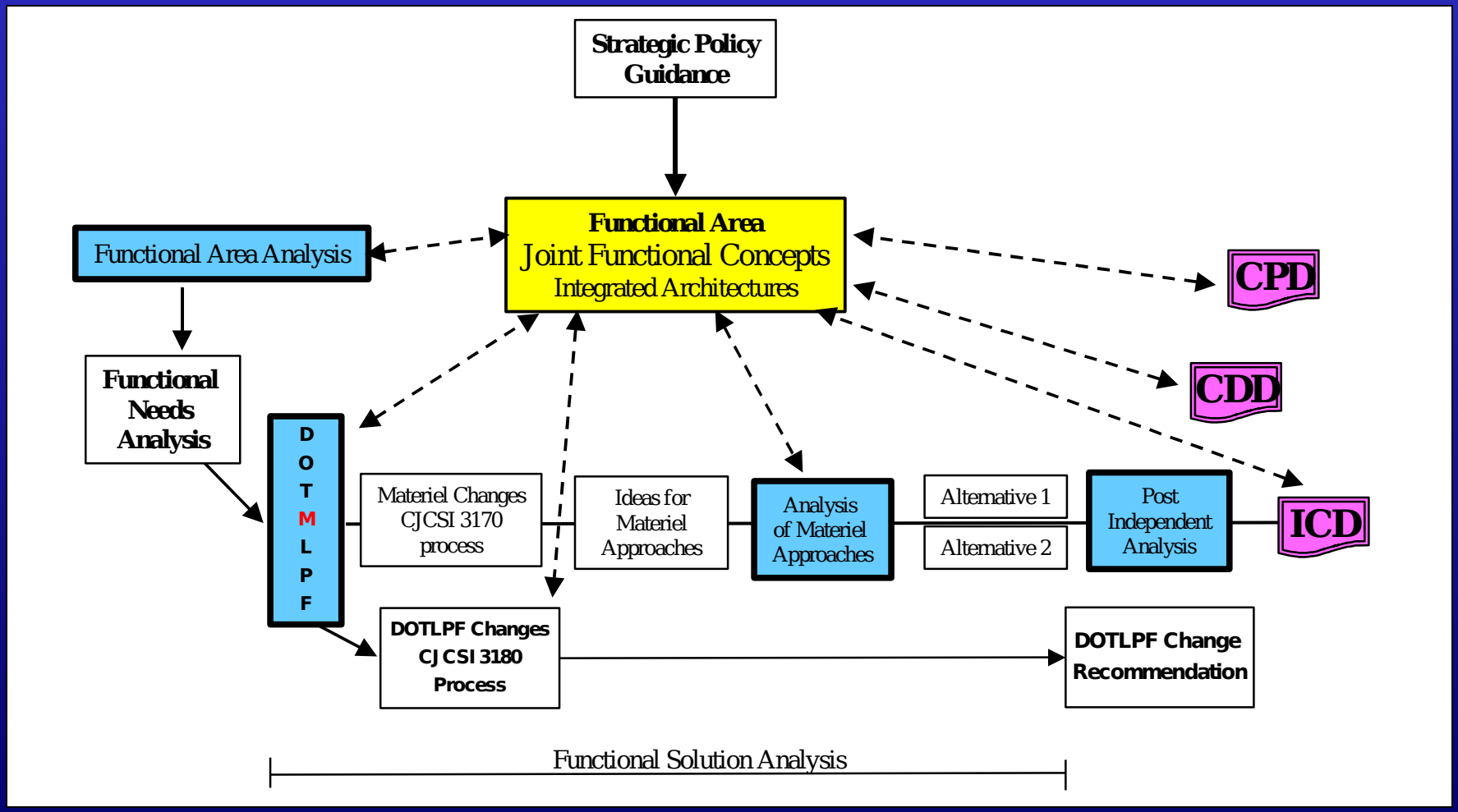
System Architecture

Plan

Train

Op & Supt

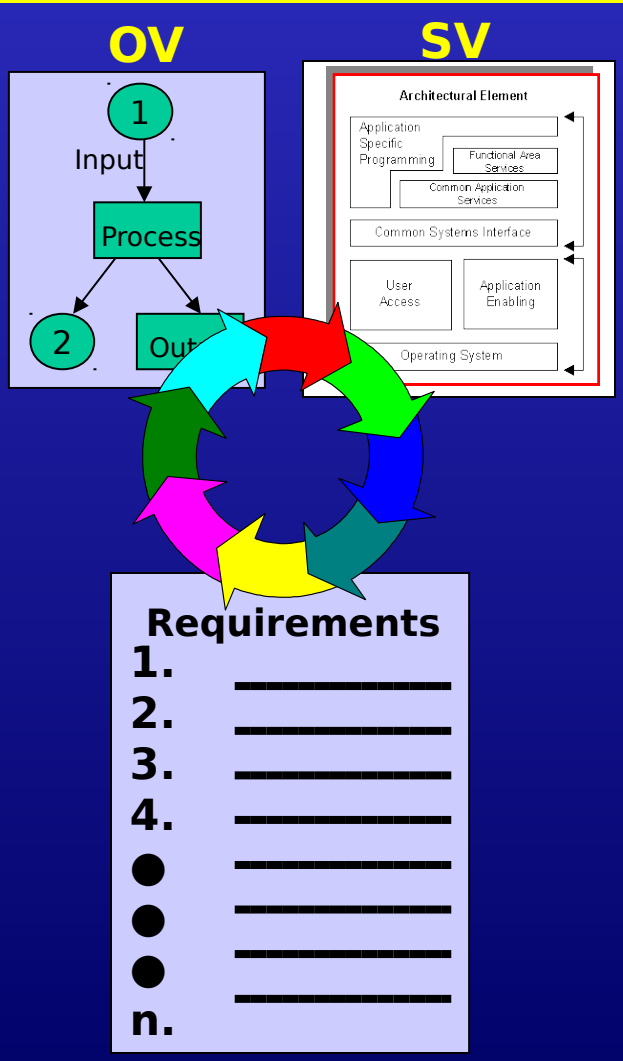
Requirements: JCIDS Input



Source: CJCSM 3170.01M FLAG STAFFING DRAFT April 2003

OV, SV and Requirements

Defense Acquisition University



- OV drives functional requirements
  - Linkage between OV and functional requirements must be maintained as requirements are allocated to modules and increments.
- SV supports OV
  - Scope and scale of functions required by OV must be supported by the systems architecture
- SV bounds functional requirements
  - Defines the manner in which requirements must be implemented
  - Establishes “enterprise rules” to govern interaction among elements
- This is an iterative and dynamic process that requires active and persistent management

Requirements Management

- Once OV and SV are established, requirements are (somewhat) bounded
  - OV describes what must be done, how, with whom
  - SV defines how the system is implemented
  - Requirements may be elaborated and re-prioritized within these constraints
  - Lifecycle costs are largely set (though perhaps not known)
- EA recognizes that operational needs will change
  - But these changes need to be validated against and reflected in the OV

Requirements Management

Defense Acquisition University

- PMO & Architect must
  - Periodically reconvene the stakeholders to formally reassess the stated requirements, their functional and incremental allocation, and implementation priority
  - Use the OV and SV to communicate the boundaries within which the program is defined
  - Incorporate results from field use, experimentation, test and evaluation
  - Facilitate discussion, capture the evolving understanding of the problem and solution spaces
    - Demonstrate prototypes and work-in-progress to make future deliverables tangible
    - Use modeling and simulation to illustrate concepts, to elicit and clarify requirements
    - Reach consensus on CAIV and Schedule-driven choices
- This is not easy, nor is it inexpensive!

Requirements Management: Ground-rules

Defense Acquisition University

- Every requirement arises from, and thus must be mapped to the OV
- Requirement change results from either
  - Elaboration of an existing requirement, or
  - A change to the OV
- Elaborated requirements are “in scope”
  - A result of gaining a clearer understanding of the requirement
  - Has cost implications, but should be handled within program constraints
  - PMO MUST budget for requirements elaboration cost growth
- Major changes to the OV require readjustment to program resources and plans
  - Mission/doctrine changes that could not be anticipated
  - Major changes that require reexamining resources assumptions
  - This discussion must be held in concert with ALL stakeholders!

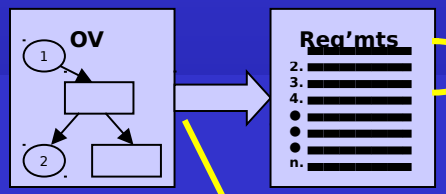
Requirements Management: Responsibilities

- User responsibilities
  - Make a good-faith investment in defining the program-level OV
  - Maintain the OV as an accurate representation of operational processes, priorities, and requirements
  - Use the OV to communicate with the acquirer and other stakeholders
  - Promptly reflect changes in doctrine, tactics, techniques, and procedures in the OV
- Acquirer responsibilities
  - Understand the User's domain via the OV
  - Honestly communicate trade-off decisions imposed by the Acquisition strategy, budget, and systems architecture
  - Identify cost, technical and schedule challenges early with User and all stakeholders

Process:
Risk Management

EA Process Overview

(Single Iteration)



Requirements Management Process

Risk Management Process

You Are Here

Cost

Risk

Hi

Low

SPIRAL

To Increment "n+1"

Feed-back

Production Process

System Architecture

Increment "n"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Increment "n+1"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Deliver

Delivery, Support & Feedback Process

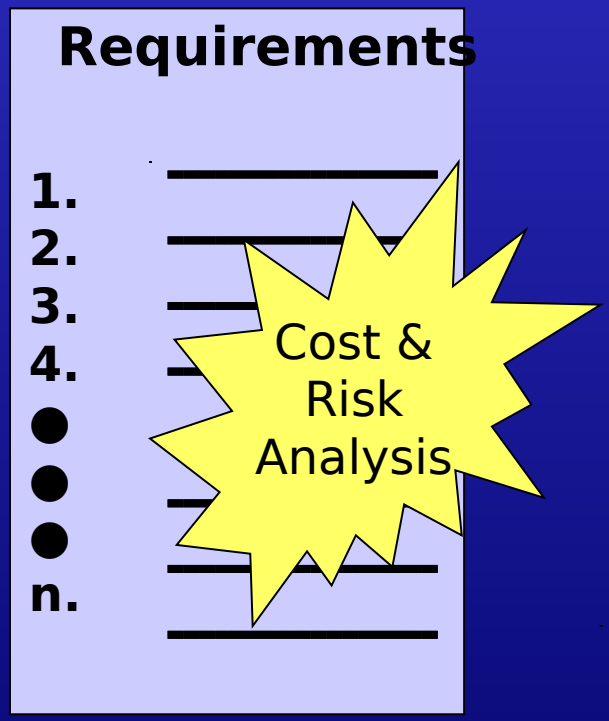
System Architecture

Plan

Train

Op & Supt

Assess Cost & Risk

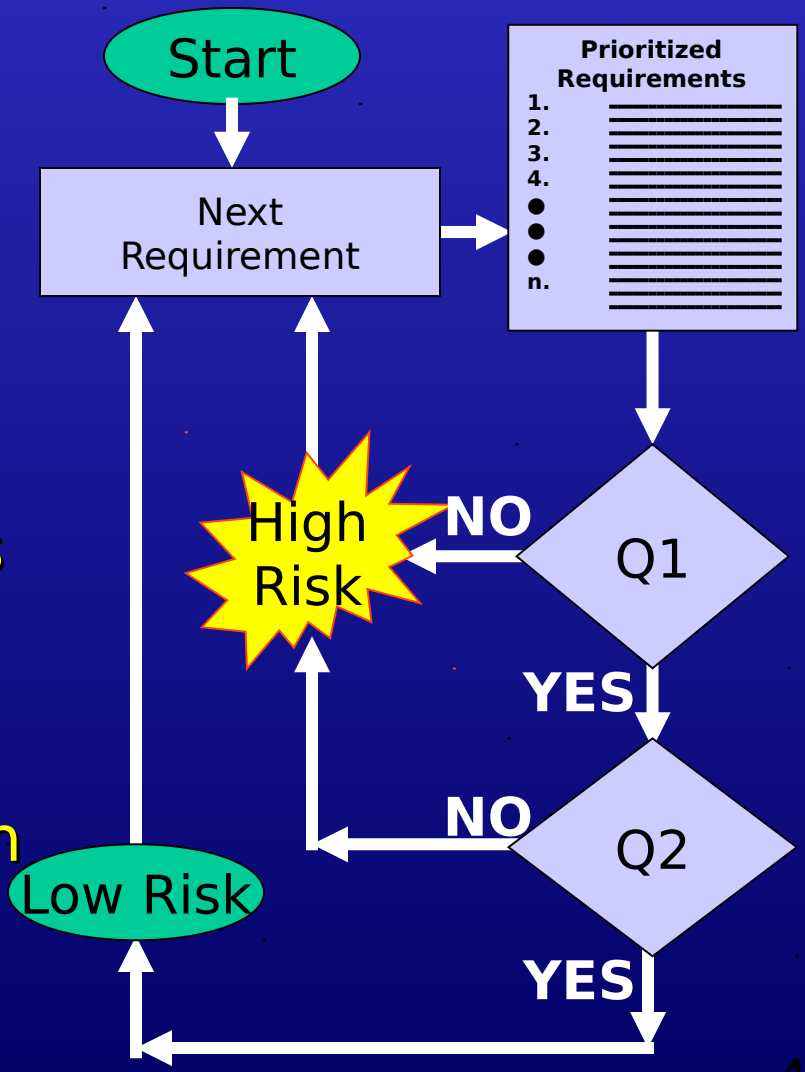


- Conduct cost and risk analysis
  - Of each requirement and functional group
  - To determine understanding of the requirement and technical maturity of potential solutions
  - To identify further dependencies
  - To provide input to budget & resource planning
  - To provide input to the Technology Development Strategy (new 5000 requirement)

Requirements Risk Assessment

Defense Acquisition University

- For <u>EACH REQUIREMENT</u> You must answer 2 fundamental questions:
  1. Do I understand what is required?
  2. Does a solution exist?
- Answer “NO” or “I’m not sure” to Q1 or Q2 means HIGH RISK (until proven otherwise)
- High Risk requirements demand special attention before implementation



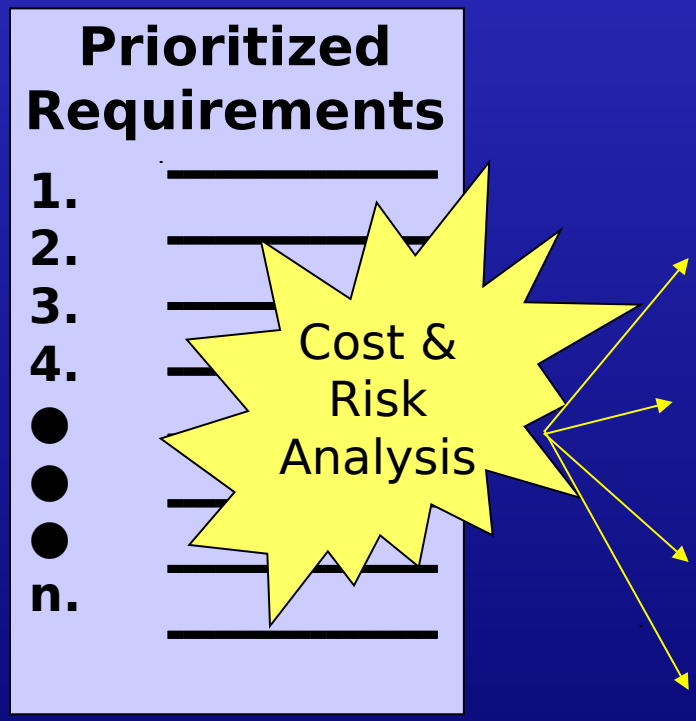
Requirements Cost & Risk

Defense Acquisition University

- Apply standard cost estimating methodologies to requirements deemed “Low Risk”
- “High Risk” requirements require further scrutiny
  - What is the nature of the risk?
    - Poorly-articulated or understood requirement statement?
    - Current technology incapable of meeting specified performance?
  - What is the “magnitude” of the risk?
    - Do the Users know what they want?
    - How big is the technology gap?
- Risk mitigation scope and cost will depend upon this analysis
- Risk increases the expected value and variance for cost and schedule

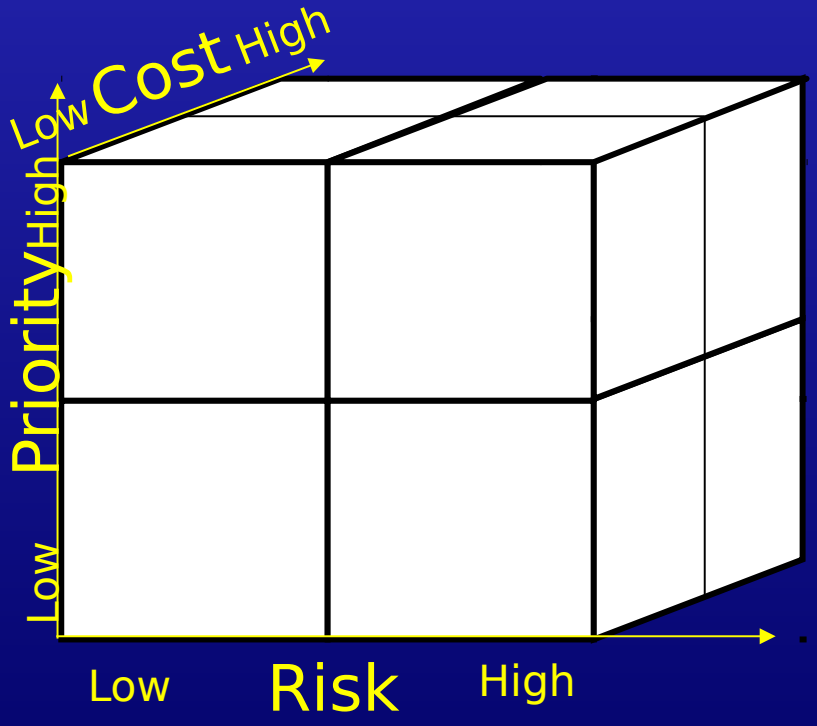
Assess Cost & Risk (continued)

Defense Acquisition University



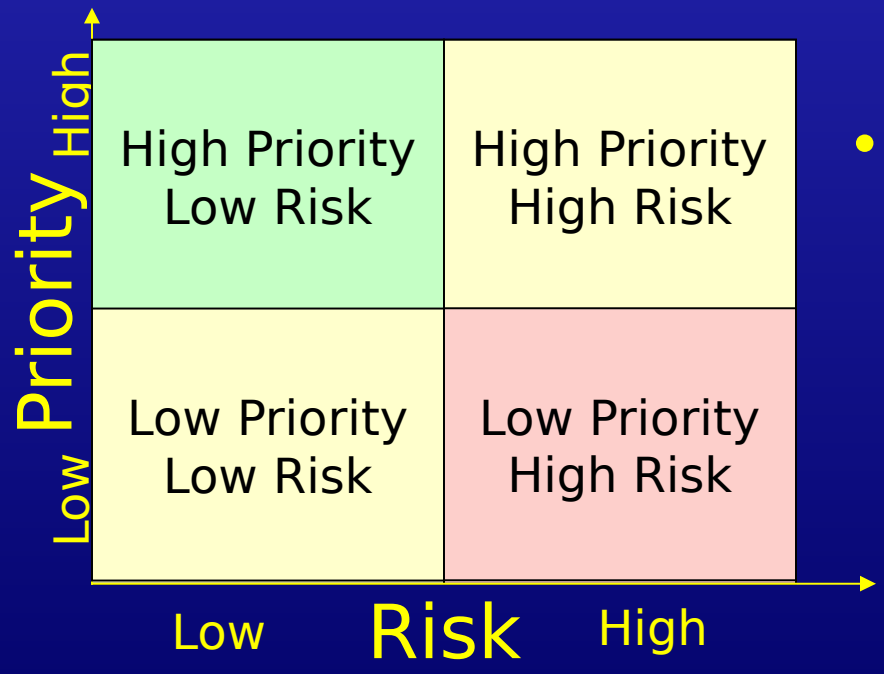
- Users, through OV and domain modeling, establish the priority for each requirement
- PMO analysts establish initial risk and cost attributes for each requirement
- These priority, cost, and risk assessments are preliminary, and are used for planning
- Continuous, proactive risk/cost management ensures affordability of build plan

Assess Cost & Risk (concluded)



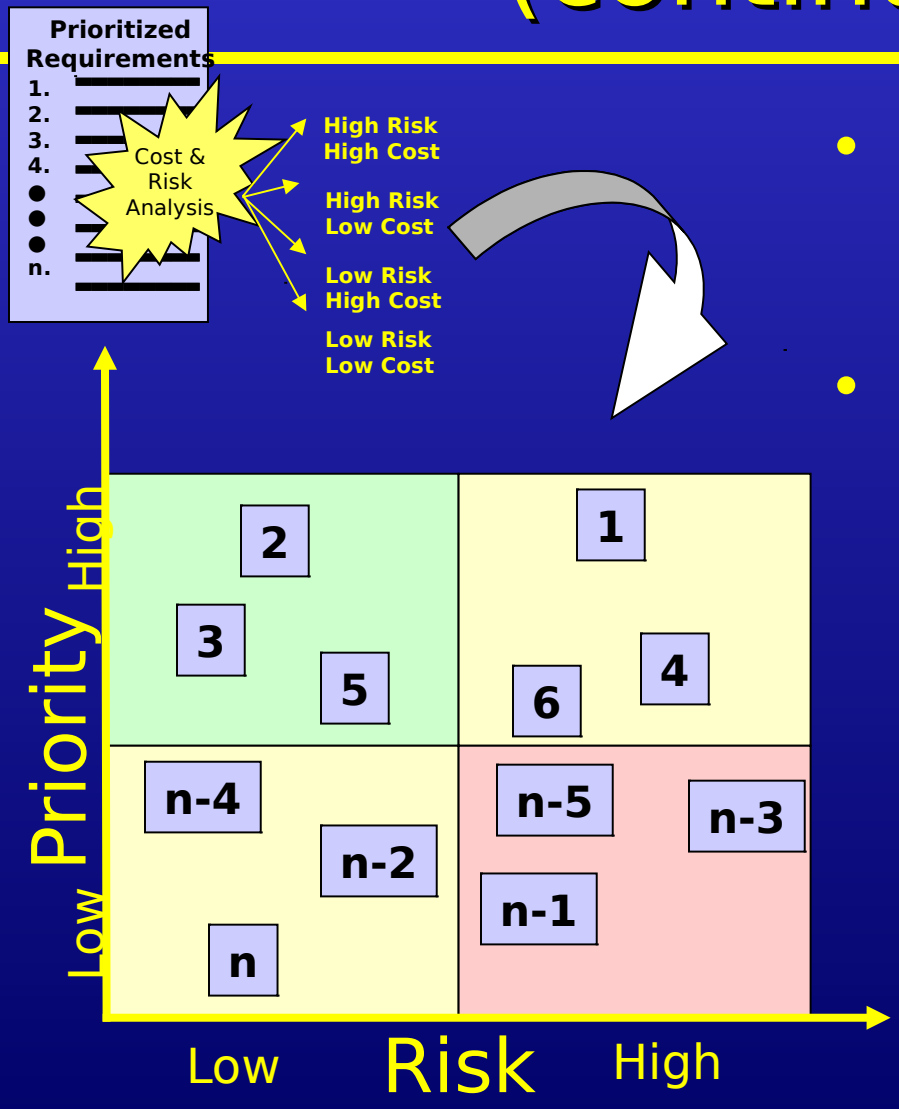
- Each requirement will have attributes of priority, cost, and risk
- Attributes can be expressed in a matrix
- Note that risk and cost are generally correlated

Sort by Priority & Risk



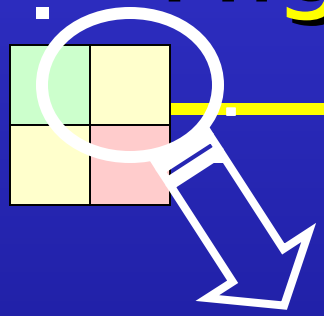
- Since cost and risk tend to be correlated, the 3D matrix can be simplified into a 2D priority vs. risk matrix
- Risk/Priority matrix provides a guide to
  - implementation sequence
  - risk management strategies
  - resource allocation priorities

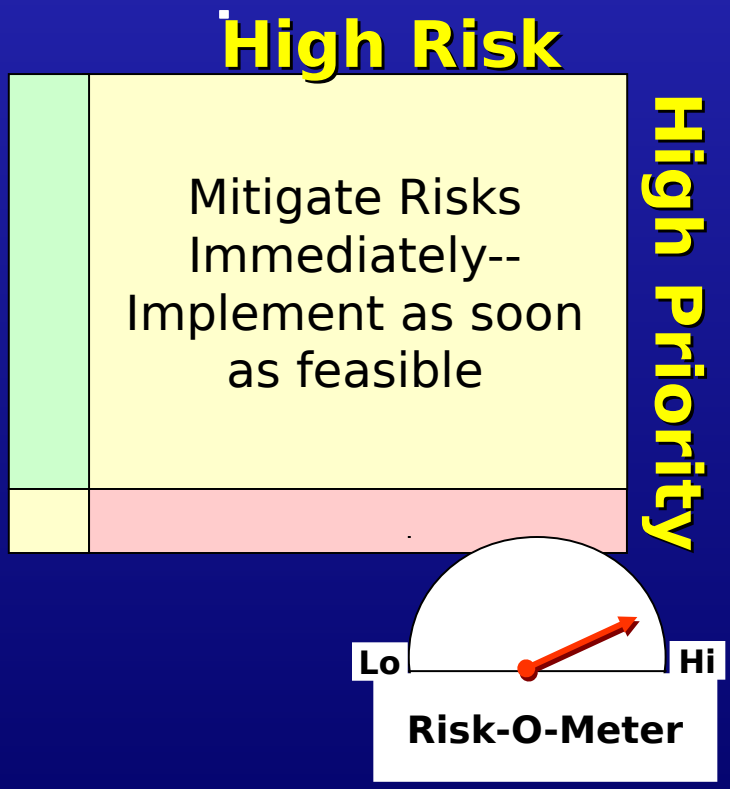
Sort by Priority & Risk (continued)



- Each requirement is mapped to the risk/priority matrix
- Use this framework to
  - Allocate requirements to increments
  - Develop risk management strategies
  - Facilitate resource planning
  - Conduct tradeoffs with Users & other stakeholders

High/High: Mitigate Risks



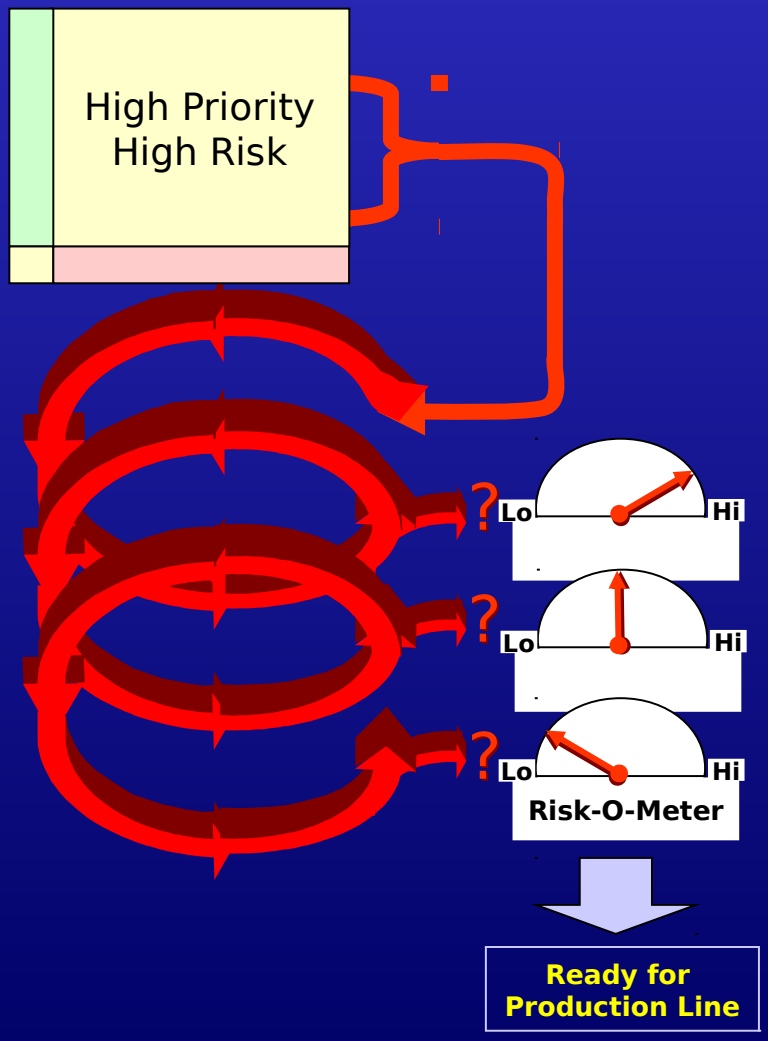


- For **High-Priority/High-Risk** requirements
  - Develop mitigation strategy based upon nature, magnitude of risk (reference previous cost/risk analyses)
  - Do not defer simply because risk is high;

-BUT-

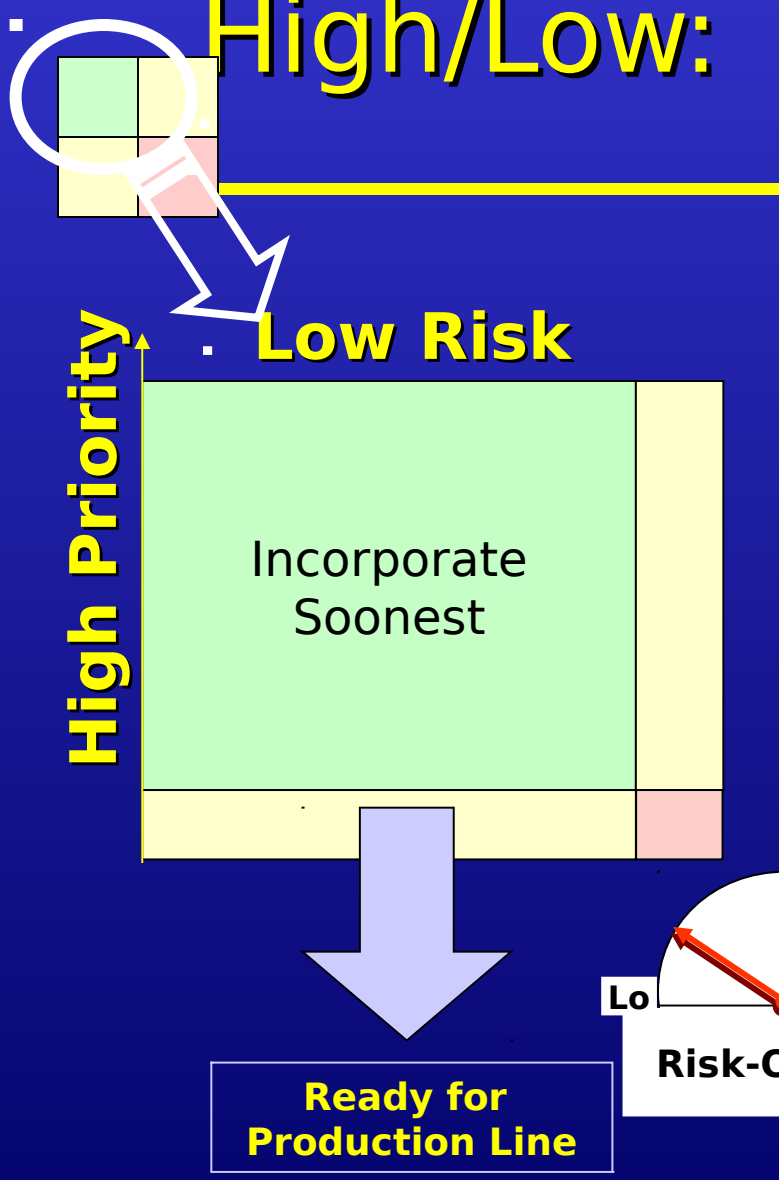
  - Do not implement until risk is reduced

Mitigate Risks (continued)



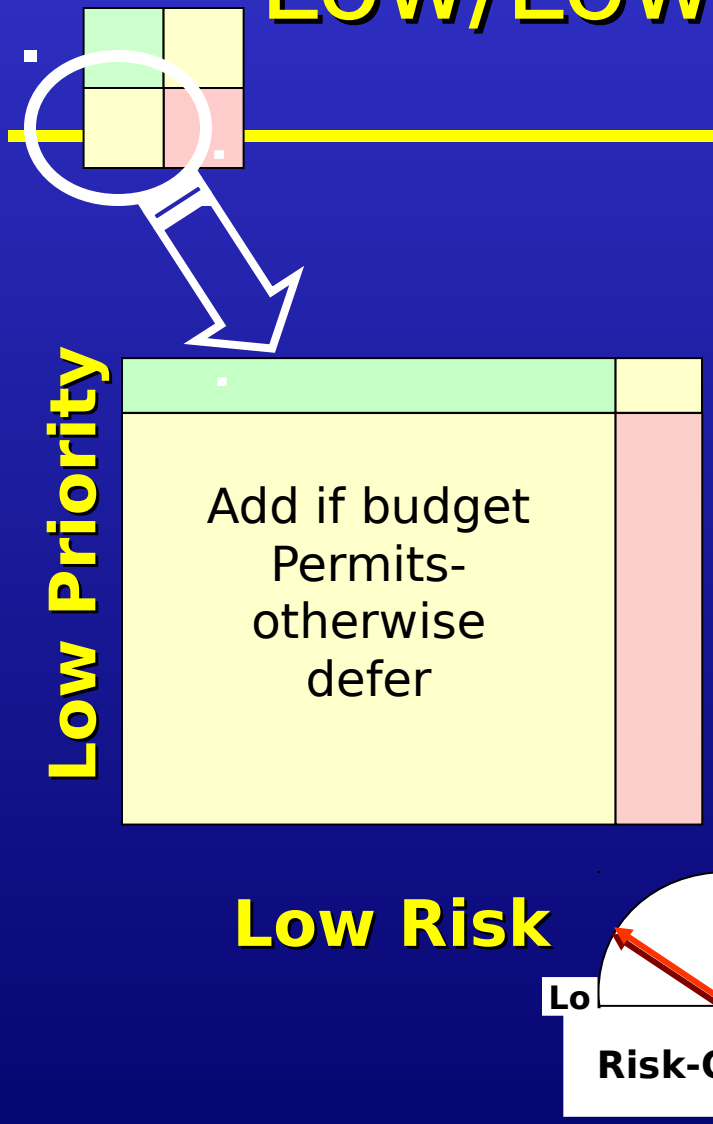
- Risk mitigation approach depends upon nature of risk
  - poorly-specified requirements
  - technical immaturity
- Spiral Development a useful tool
  - Rapid prototyping with User to resolve requirements ambiguities
  - Iterative prototypes to resolve technical maturity issues
- Keep prototyping activities focused
  - Establish limits on cost and duration
  - Tailor output to feed production line
  - Resist temptation to field prototypes!
- Exit Criterion:
  - Low Implementation Risk

High/Low: Go to Production



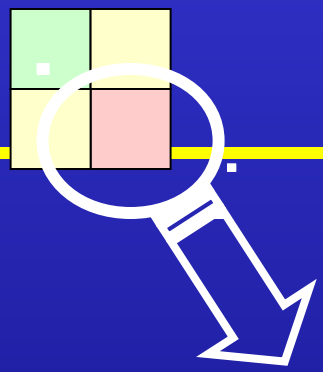
- For **High-Priority/Low-Risk** requirements
  - Allocate to increment based on
    - Priority
    - Resource and other constraints
    - Dependencies with other requirements
  - Don't devote entire budget to Low Risk requirements
    - High Risk "Gotchas" are out there waiting to bite you!

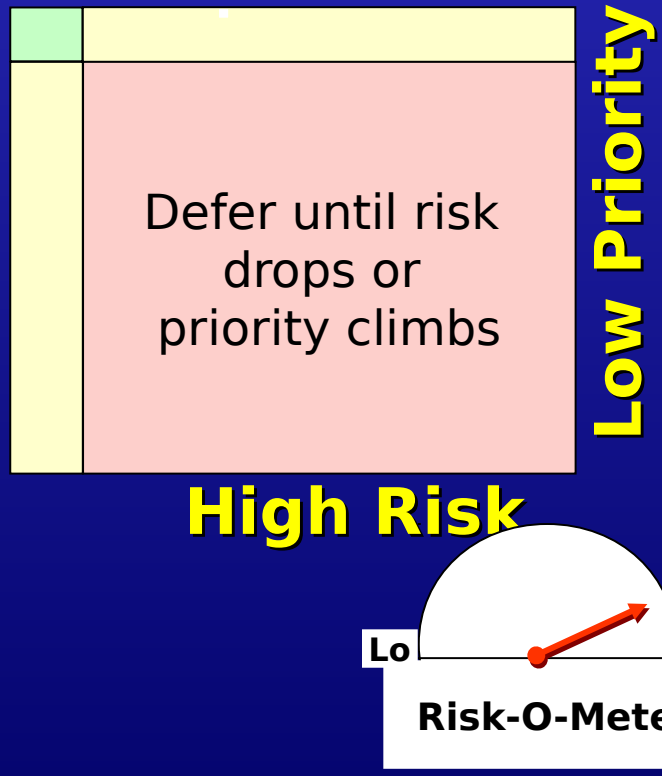
Low/Low: Now or Later



- Don't commit to production simply based upon low risk
- User Priority should drive production sequence
- Low priority-low risk requirements should be added only after higher priority requirements have been attended to

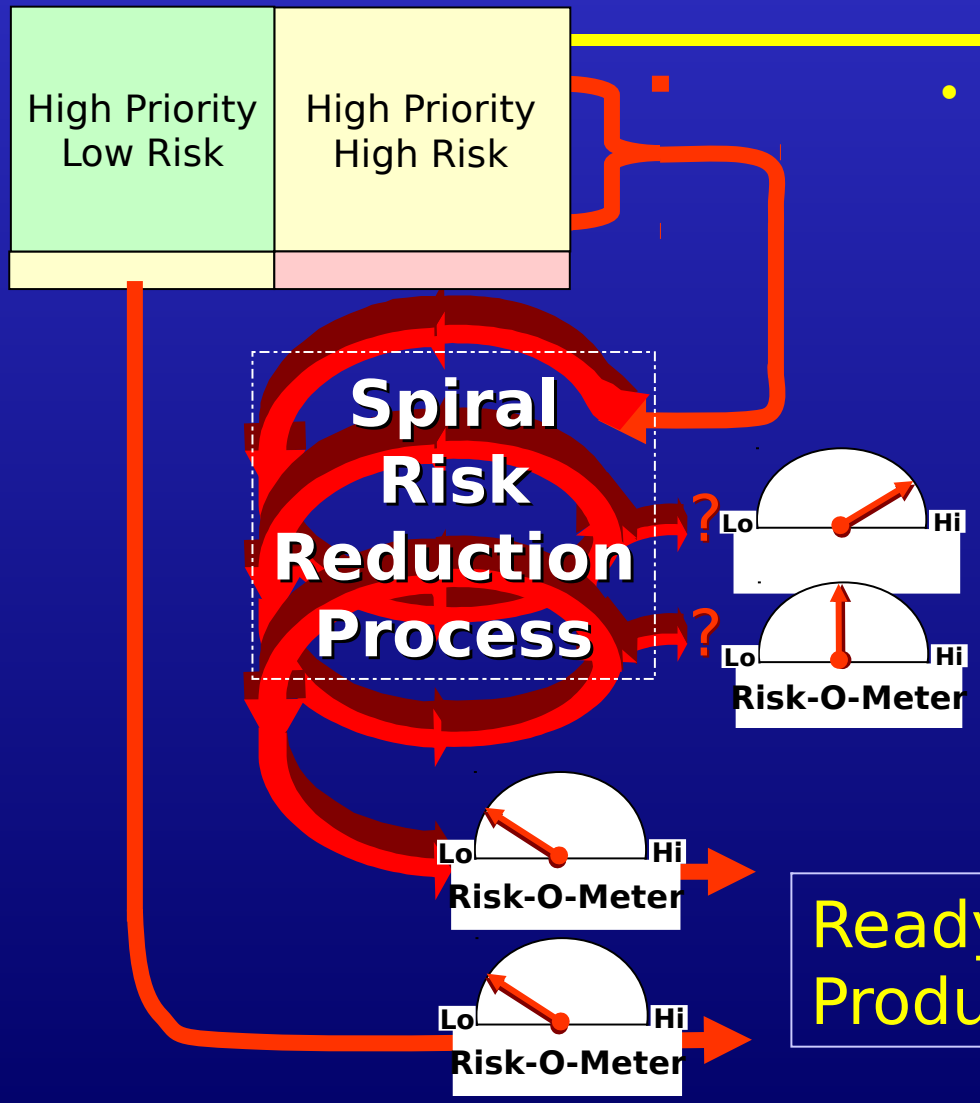
Low/High: Defer





- Low Priority/High Risk requirements should be deferred, but not discarded
- Priorities and risks are relative, and will change over the implementation period
  - As top priorities are satisfied, the lower priorities will rise
  - As technology progresses, risky requirements will become feasible
  - As new threats emerge, low priority enhancements may become essential
  - As doctrine and policy evolve, so will the OV, giving rise to new, or re-prioritized requirements

Production Entry Criteria

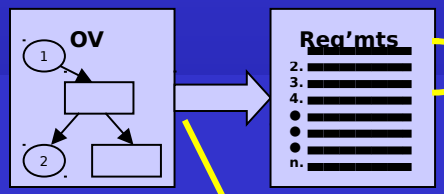


- Transfer only low-risk high-priority tasks to the production line
  - Addresses Users' most pressing needs
  - Allows efficient production processes to be used
  - Enables predictable increment deliveries (important to Users)

Process: Production

EA Process Overview

(Single Iteration)



Requirements Management Process

Risk Management Process

Cost

Risk

Hi

Low

SPIRAL

To Increment "n+1"

Feed-back

Production Process

You Are Here

System Architecture

Increment "n"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Increment "n+1"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Delivery, Support & Feedback Process

System Architecture

Deliver

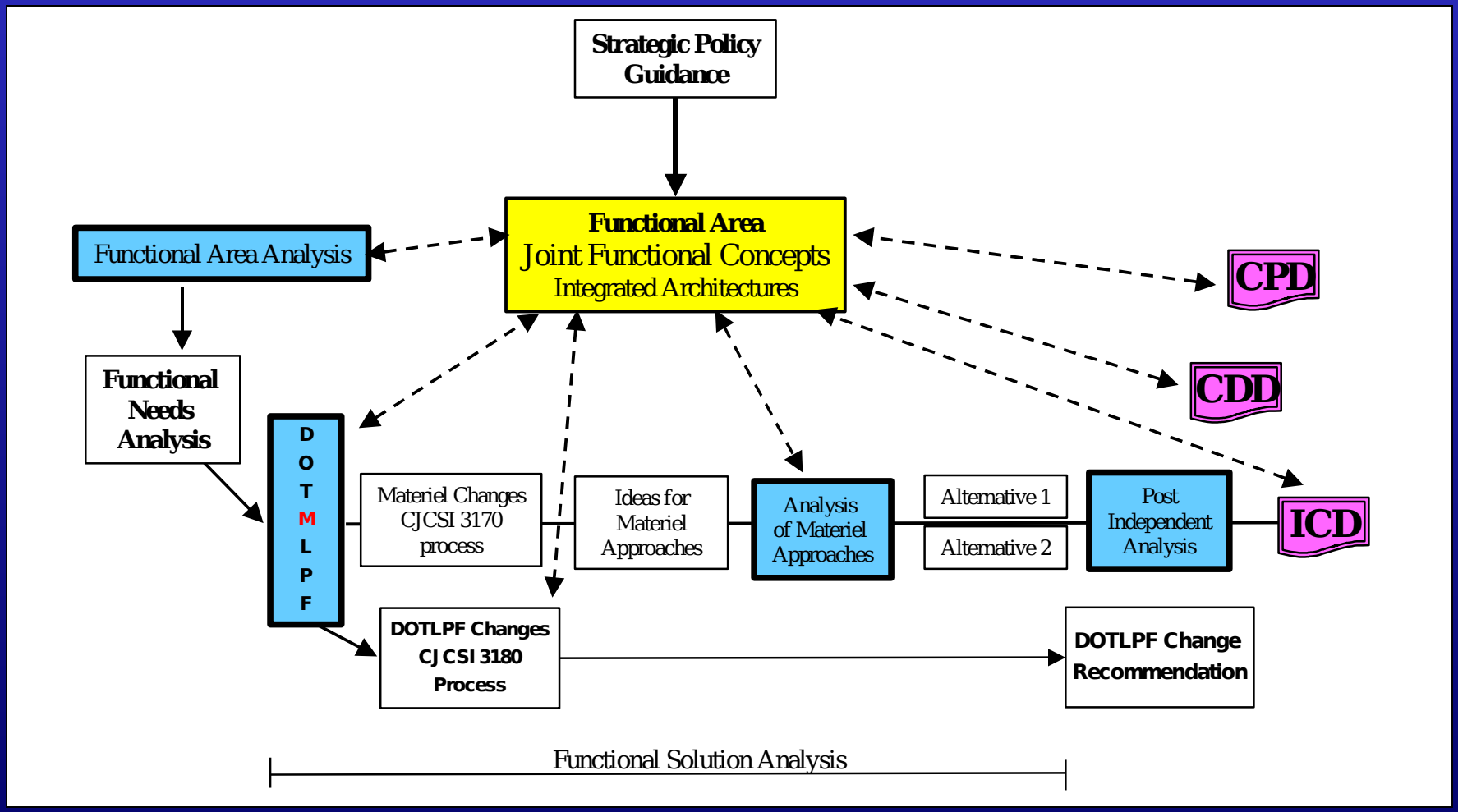
Plan

Train

Op & Supt

Production: JCIDS Input

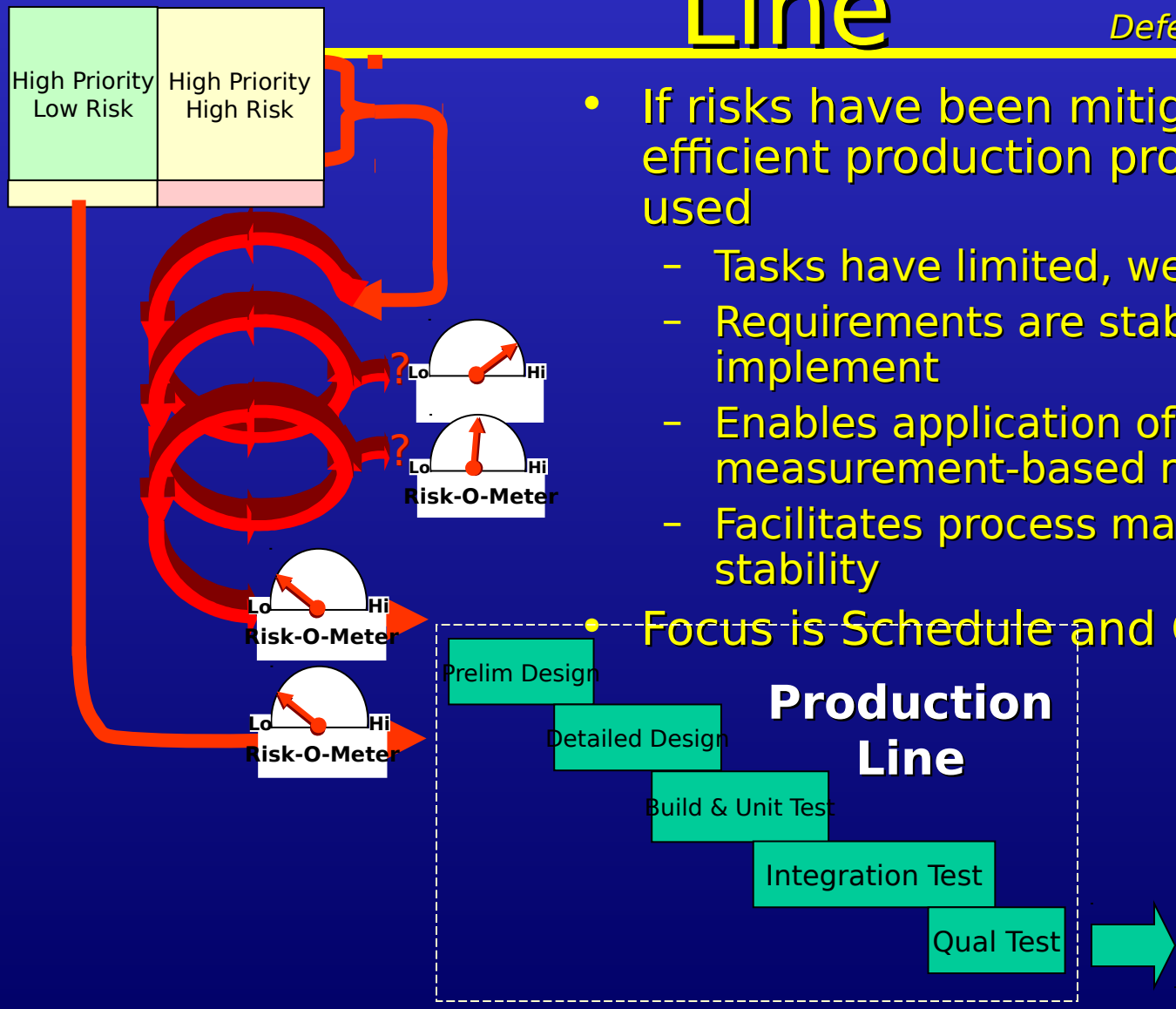
Defense Acquisition University



Source: CJCSM 3170.01M FLAG STAFFING DRAFT April 2003

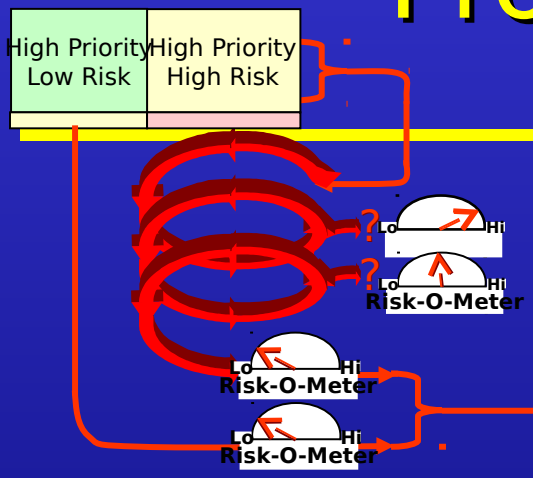
Concept: The Production Line

Defense Acquisition University



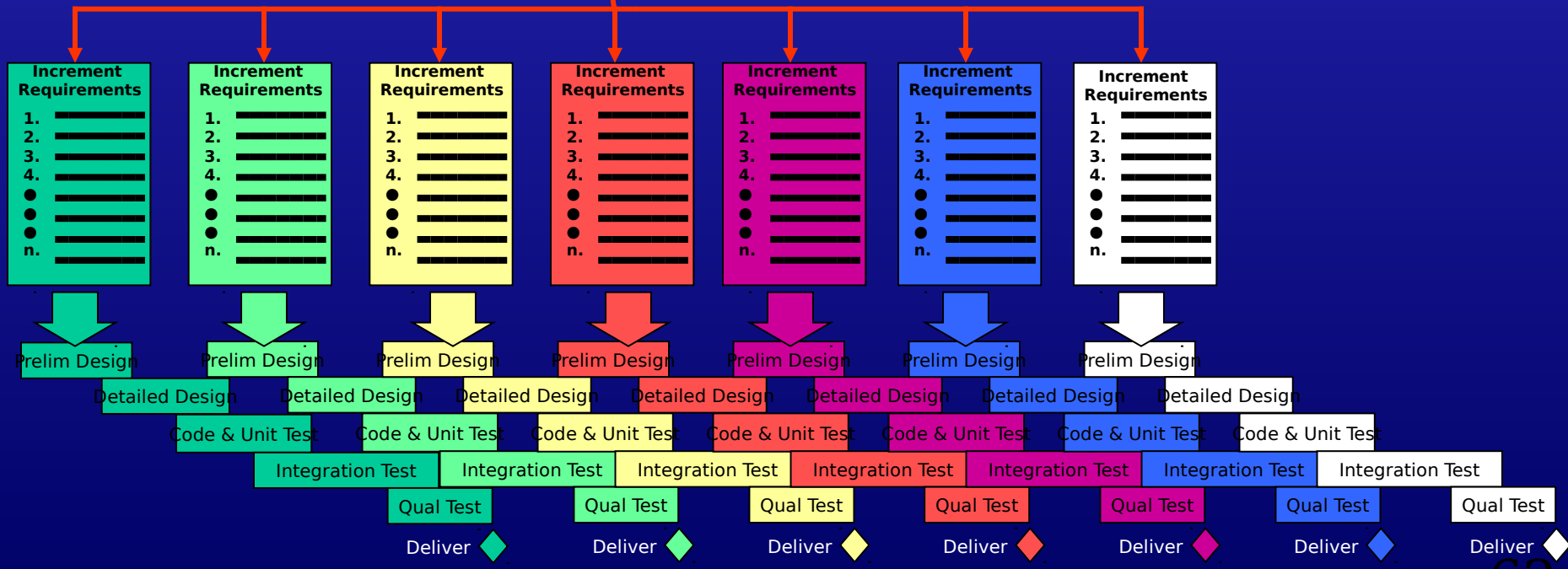
- If risks have been mitigated, highly efficient production processes can be used
  - Tasks have limited, well-defined scope
  - Requirements are stable, low risk to implement
  - Enables application of CM, EVM and other measurement-based management tools
  - Facilitates process maturity, workforce stability
- Focus is Schedule and Cost predictability

Production Process



Defense Acquisition University

- Series of incremental builds
- Incorporate high priority low risk requirements
- Efficient development processes
- Reliable product
- Predictable cost and schedule

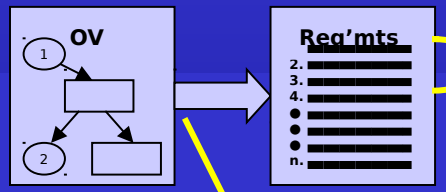


Process: Delivery, Support & Feedback

EA Process Overview

(Single Iteration)

Requirements Management Process



Risk Management Process

Cost

Risk

Hi

Low

SPIRAL

To Increment "n+1"

Feed-back

Production Process

System Architecture

Increment "n"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Increment "n+1"

Prelim Dsgn

Detail Dsgn

Code & UT

Integ'n Test

Deliver

Delivery, Support & Feedback Process

System Architecture

You Are Here

Plan

Train

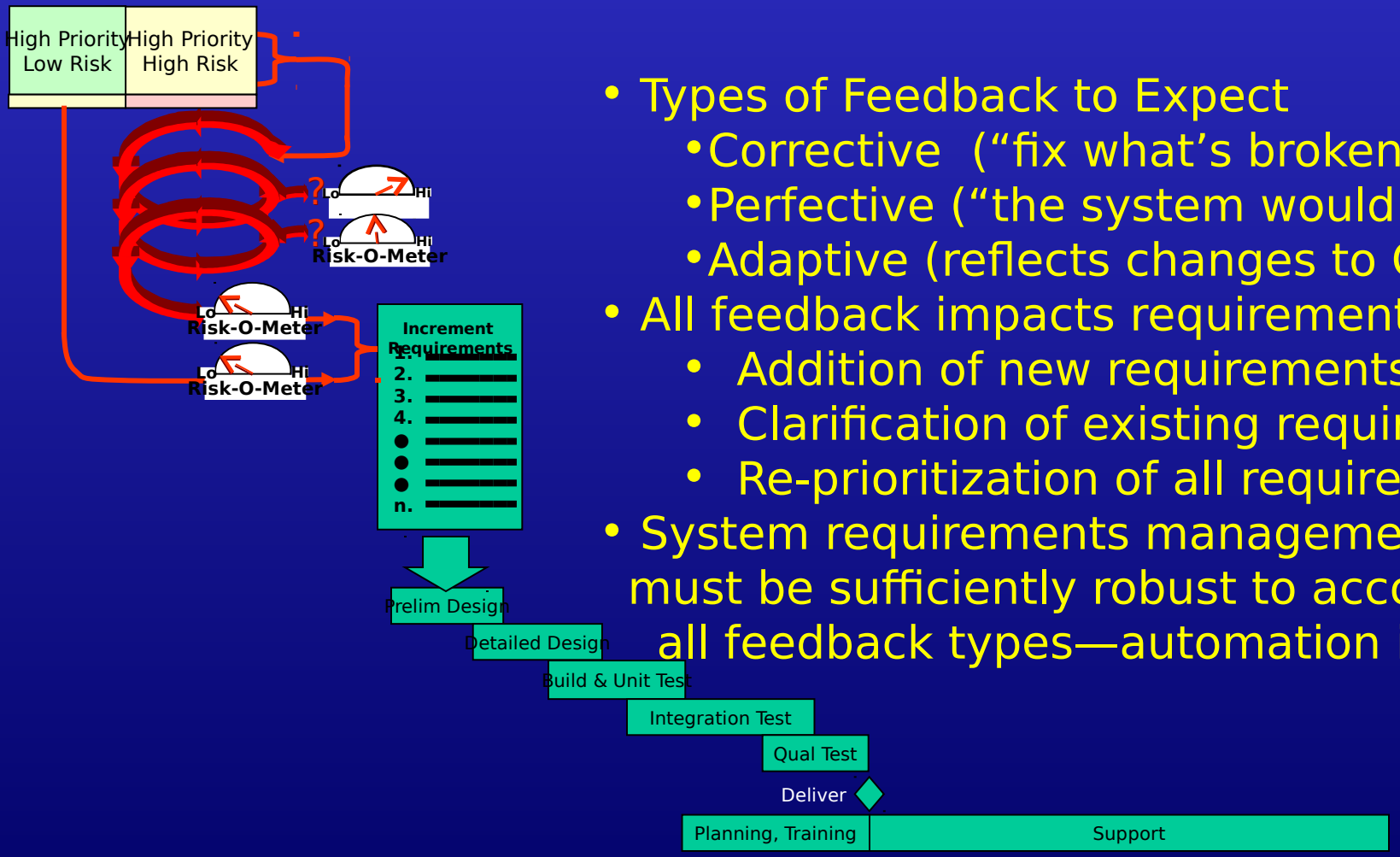
Op & Supt

Key Expectations

- Deliver on-time with promised capability
  - Maintain credibility of acquisition process
- Support what is delivered
- Establish integrated corrective action & requirements management process
  - Distinction between “development” and “support” is not as meaningful in EA
  - Establish regular feedback, requirements development, validation, prioritization meetings
  - Leverage training, testing, experimentation opportunities to gather “ground truth” data

Delivery, Support & Feedback

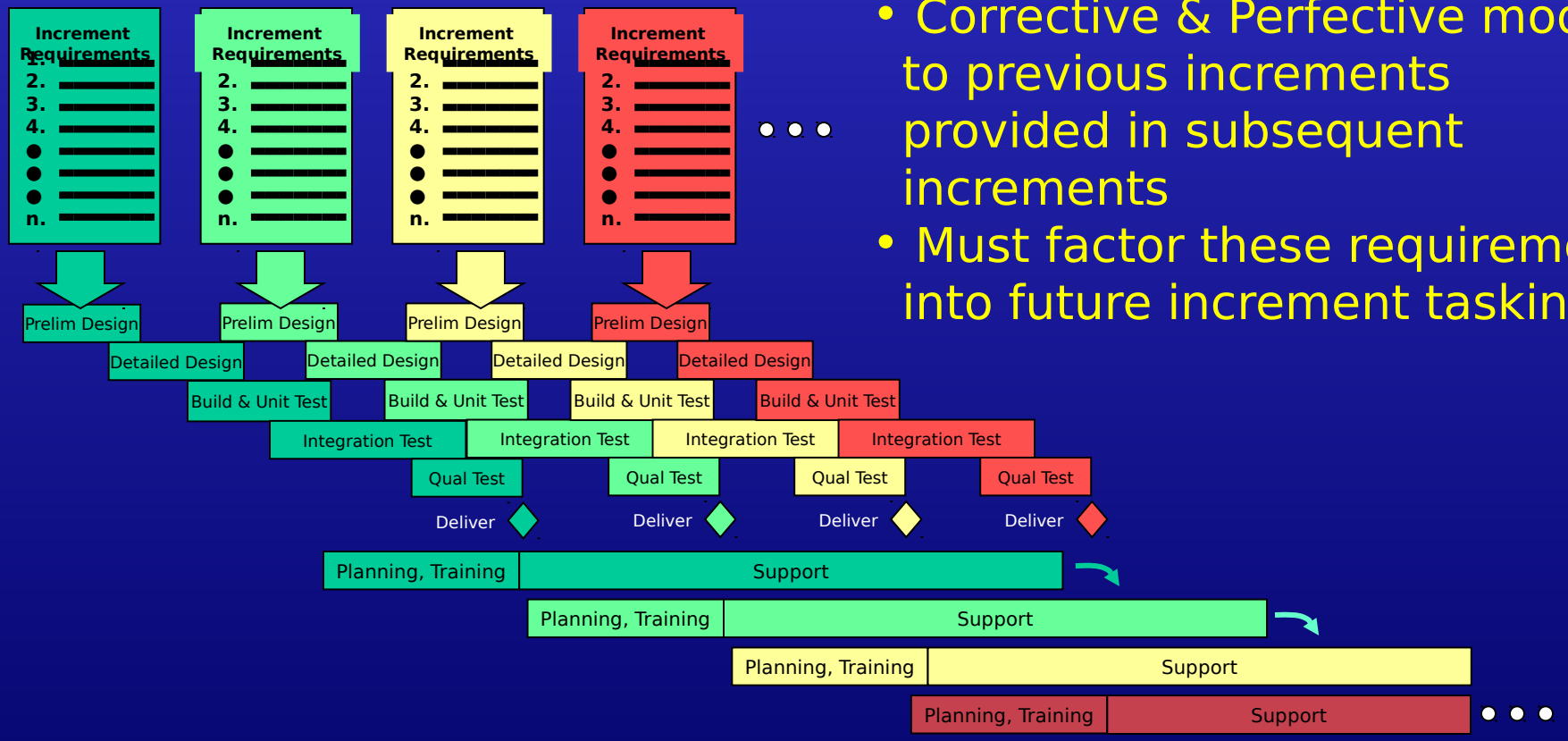
Defense Acquisition University



- Types of Feedback to Expect
  - Corrective ("fix what's broken")
  - Perfective ("the system would be bett
  - Adaptive (reflects changes to OV)
- All feedback impacts requirements
  - Addition of new requirements
  - Clarification of existing requirements
  - Re-prioritization of all requirements
- System requirements management proce must be sufficiently robust to accommoda all feedback types—automation is a mus

Supporting Increments

Defense Acquisition University



- Corrective & Perfective mods to previous increments provided in subsequent increments
- Must factor these requirement into future increment tasking

Acquisition Process Summary

Defense Acquisition University

- Acquirer must collaborate with user in program-level architecture and requirements definition
- Acquirer must stay engaged with user throughout product lifecycle
- Incorporate business, support, and test planning in the architecture phase
- Establish and distribute executable program-level systems architecture as first “deliverable”
- Implement functionality in order of operational priority using OV as reference
- Mitigate risk of high-priority requirements using spiral development
- Send only low-risk high-priority requirements to production line
- Integrate test and support early and throughout increment development
- Institute formal support & feedback over lifecycle

Special Interest Items

Contracting Implications

Logistics Implications

Test & Evaluation Implications

Contracting Implications

Contracting For EA

Defense Acquisition University

- Establishing Fundamental Assumptions
- Source Selection
- Activities
- Contract Types

Establishing Fundamental Assumptions

Defense Acquisition University

- PMO must establish whether the acquisition is
  - A continuous acquisition within which increments of functionality will be developed
  - A series of discrete acquisitions
- Determinants:
  - Size: Major acquisitions require separate milestone decisions for each increment; which prevents continuity of contract between increments
  - Domain: COTS-based acquisitions that have robust commercial analogs may have open, modular domain architectures that enable piecewise acquisition

Source Selection For EA

- Dominant considerations for selecting EA developers
  - Architecture development/management
    - Ability to develop and support an open, scalable, secure & robust architecture
    - May be of less importance in COTS-based programs where robust, modular, open architectures exist
  - Process management
    - Ability to manage the complex EA process
  - Vendor, subcontractor management
    - Ability to impose interoperability requirements on functional developers
  - Development, integration, & support of functional modules

Contracting For EA

Defense Acquisition University

- Activities
  - System program & process management
    - Maintaining close contact with stakeholders
    - Maintaining tight control of the processes and players
  - System architecting, engineering and integration
    - Maintaining control of the architecture
    - Imposing design standards on functional elements
  - Iterative requirements development
  - Continuous risk management & technology transition
  - Parallel development of individual increments
  - Repeated deployment/delivery of individual increments
  - Continuous support and maintenance of delivered increments
  - Incremental & integrated test & evaluation

Contracting Summary

- Source Selection should be based upon process capabilities more than technical or cost (my opinion, based upon)
  - Program disasters are usually related to management shortcomings, not technical obstacles (Nov 2000 DSB report on DoD Software)
  - You can hire technical talent, but you can't buy process maturity after contract is awarded
- Consider architecture development as a separate activity
- Develop/maintain systems architecture as a distinct entity from functional modules
  - Consider maintaining a "systems architect" for the program
- Modular contracting is essential

Logistics Implications

Logistics Implications of EA

- Time-phased requirements mean that the capabilities of delivered items will evolve over time
- Different capabilities mean different configurations
- Different configurations will make logistics support more complex, and thus more costly
- Look out for interoperability issues!
  - Different units with different versions of the same equipment may not be able to interoperate
  - This may impact support resources

Logistics Role in EA

- To reduce support costs, supportability issues must be addressed early
  - In an IPPD environment
  - To influence architecture and design decisions
    - Open vs. proprietary architecture
    - Modular or tightly-coupled design
  - To influence support concept
    - CLS, PBL, Organic
    - Support multiple CI's or retrofit

Architecting for supportability

- Modular, scalable, open systems architecture minimize the support cost of multiple CI's
  - Core functionality allocated to the architecture
  - Mission-specific functionality allocated to modular elements
  - Functional modules can be changed without extensive impact to architecture or other elements.
  - Support for core will be similar across configurations
  - Unique support requirements limited to functional modules
  - Retrofit/upgrade cost is minimized

Logistics Summary

- Logistics support issues must be addressed early
- Investment in a supportable architecture is vital to minimize support costs
- Critical choices must be made
  - Multiple CI's or retrofit to single CI
  - Organic or Contractor logistics support
  - Can PBL work for your product?

Test & Evaluation Implications

T&E Implications of EA

- Time-phased requirements mean that the capabilities of delivered items will differ from increment to increment
- Each increment must be tested to demonstrate achievement of requirements
- Integrated system must demonstrate operational effectiveness and suitability

T&E's Role in EA

- T&E issues must be addressed early
  - In an IPPD environment
  - To influence architecture and design decisions
    - Open vs. proprietary architecture
    - Modular vs. tightly-coupled design
  - To identify test issues and requirements
    - Full-Rate-Production decision & Beyond LRIP Report
    - Initial Operational Capability point
    - Live Fire Test & Evaluation requirements
    - Identify any waivers required
  - To influence test concept
    - Integrating development and operational test
    - Operational test intervals
    - Degree of regression testing required
  - To influence resource decisions
    - Investment in integrations and test facilities
    - Ensure adequate test resources available

Architecting for Testability

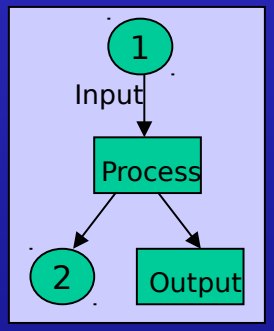
Defense Acquisition University

- Modular, open systems architecture may reduce the test impact of EA
  - Core/common functionality allocated to the architecture
  - Mission-specific functionality allocated to modules
  - Functional modules can be added/changed without extensive impact to architecture or other elements.
  - Regression test (re-verifying previously-tested functionality) from increment to increment may be reduced
- Validation of the architecture via modeling and simulation is crucial
  - Refine test concepts
  - Demonstrate architectural modularity with respect to regression testing
  - Help define test intervals and resources required

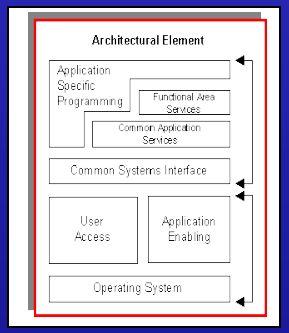
Integrate T&E Considerations Into Requirements Process

Defense Acquisition University

OV



SV



Requirements

1. _____
2. _____
3. _____
4. _____
● _____
● _____
● _____
n. _____

- Linkage between OV and functional requirements defines operational effectiveness criteria for OT
- This linkage must be maintained as requirements are allocated to modules and increments.
- The criteria and rationale for each requirement feeds the test plan development process

T&E Summary

- T&E issues must be addressed early
- Investment in a modular architecture is vital to minimize T&E costs
- Use M&S to clarify test, resource issues
- Critical choices must be made
  - Integration/Deconfliction of DT and OT
  - OT intervals; regression testing required
  - LFT&E, BLRIP, IOT&E, FOT&E requirements
    - Obtaining D,OT&E and congressional buy-in
  - Program, Test Agency resources allocated to the program

In Conclusion: Good News/Bad News…

- Good News:
  - Opportunities for innovation
  - Permission to be flexible
- Bad News:
  - Guidance & understanding is still evolving
  - Considerable challenges remain
- Bottom Line: Acquisition know-how, experience, integrity, and courage will be at a premium